

**NASA
Technical
Paper
3053**

January 1991

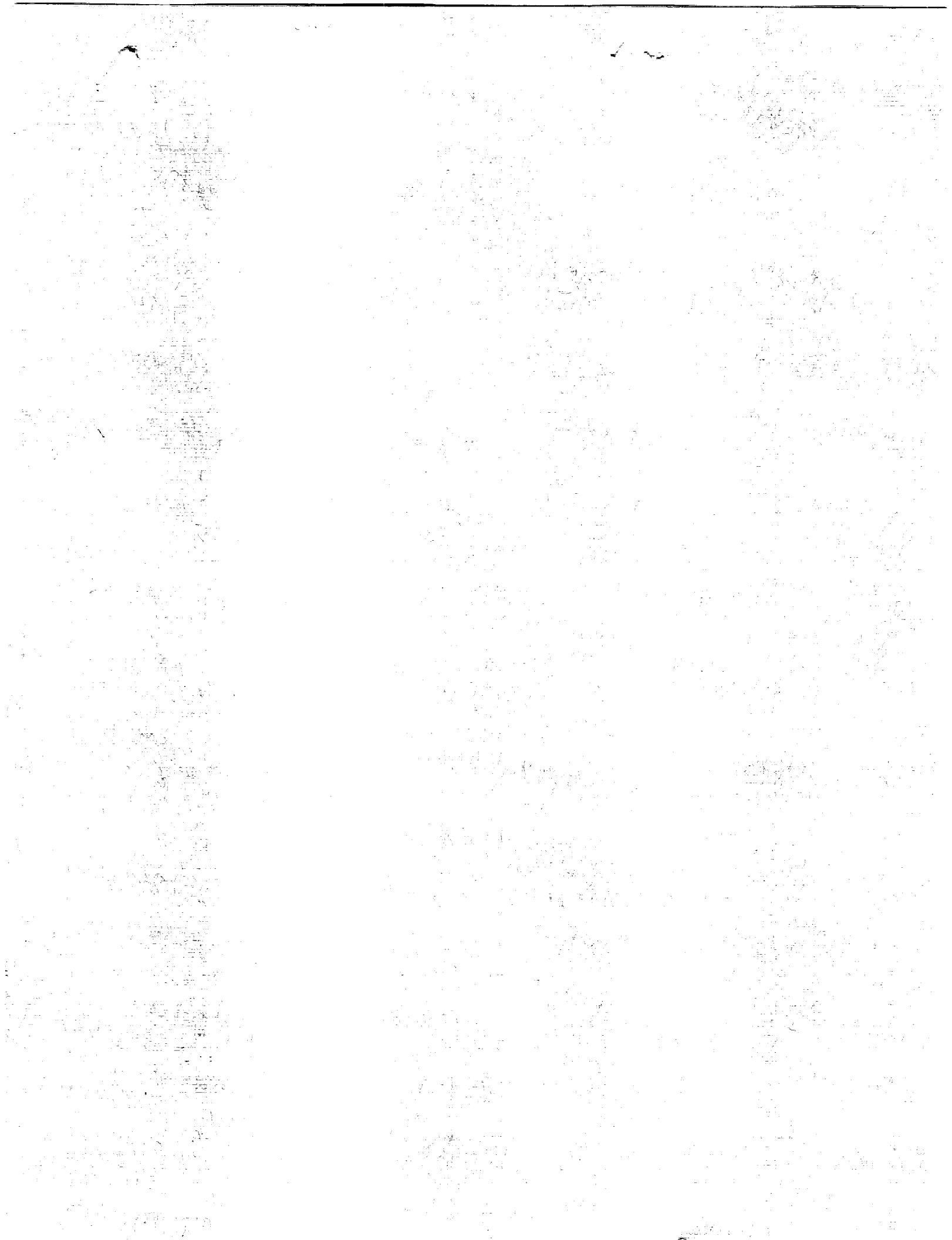
**J-85 Jet Engine Noise Measured
in the ONERA S1 Wind Tunnel
and Extrapolated to Far Field**

**Paul T. Soderman,
Alain Julienne,
and Adolph Atencio, Jr.**

AMES
IN-71
1583

P181

NASA



**NASA
Technical
Paper
3053**

1991

**J-85 Jet Engine Noise Measured
in the ONERA S1 Wind Tunnel
and Extrapolated to Far Field**

Paul T. Soderman
*Ames Research Center
Moffett Field, California*

Alain Julienne
*Office National d'Études et de Recherches Aérospatiales
Châtillon (Hauts-de-Seine), France*

Adolph Atencio, Jr.
*Ames Research Center
Moffett Field, California*



National Aeronautics and
Space Administration
Office of Management
Scientific and Technical
Information Division

J-85 JET ENGINE NOISE MEASURED IN THE ONERA S1 WIND TUNNEL AND EXTRAPOLATED TO FAR FIELD

Paul T. Soderman, Alain Julienne,¹ and Adolph Atencio, Jr.

Ames Research Center

SUMMARY

Noise from a J-85 turbojet with a conical, convergent nozzle was measured in simulated flight in the ONERA S1 Wind Tunnel. Data are presented for several flight speeds up to 130 m/sec and for radiation angles of 40° to 160° relative to the upstream direction. The jet was operated with subsonic and sonic exhaust speeds. A moving microphone on a 2-m sideline was used to survey the radiated sound field in the acoustically treated, closed test section. The data were extrapolated to a 122-m sideline by means of a multiple-sideline source-location method, which was used to identify the acoustic source regions, directivity patterns, and near field effects. The source-location method is described along with its advantages and disadvantages.

Results indicate that the effects of simulated flight on J-85 noise are significant. At the maximum forward speed of 130 m/sec, the peak overall sound levels in the aft quadrant were attenuated approximately 10 dB relative to sound levels of the engine operated statically. As expected, the simulated flight and static data tended to merge in the forward quadrant as the radiation angle approached 40°. There is evidence that internal engine or shock noise was important in the forward quadrant. The data are compared with published predictions for flight effects on pure jet noise and internal engine noise. A new empirical prediction is presented that relates the variation of internally generated engine noise or broadband shock noise to forward speed. Measured near field noise extrapolated to far field agrees reasonably well with data from similar engines tested statically outdoors, in flyover, in a wind tunnel, and on the Bertin Aerotrain. Anomalies in the results for the forward quadrant and for angles above 140° are discussed.

The multiple-sideline method proved to be cumbersome in this application, and it did not resolve all of the uncertainties associated with measurements of jet noise close to the jet. The simulation was complicated by wind-tunnel background noise and the propagation of low-frequency sound around the circuit.

¹Research Engineer, ONERA (Office National d'Études et de Recherches Aéropatiales).

INTRODUCTION

In the fall of 1979, a NASA/ONERA joint research study of jet noise was conducted in the ONERA S1 Wind Tunnel at Modane-Avrieux, France. The objective of the program was to measure the near-field noise of a General Electric J-85 jet engine at flight speeds greater than those previously attained in the NASA Ames 40- by 80-Foot Wind Tunnel and to extrapolate the data to the far field in order to identify forward speed effects on the jet noise. The data were to be compared to existing flight data. At the same time, ONERA desired to know (1) if the closed test section of the S1 Wind Tunnel could be adequately treated with acoustic linings to provide the proper acoustic environment for this type of research, and (2) which of several experimental techniques properly identified the noise source locations in the jet exhaust. The source-location techniques evaluated were (1) traversing sideline microphones, (2) acoustic antennas, and (3) infrared detectors; only the first technique is described in this report. Reference 1 describes the results of the infrared detector measurements.

The J-85 engine was chosen for this study because an extensive data base has been acquired for it by independent researchers over several years. The J-85 jet noise was measured statically (refs. 2 and 3), in flight (refs. 4–7), on a low-noise train (refs. 8–10), and in the Ames 40- by 80-Foot Wind Tunnel (refs. 11–14) before the acoustic treatment of the test section. This data base was important for checking the results of the present study because, despite the satisfactory extrapolation, in a number of studies, of near field to far field jet noise from scale models (refs. 15–18), the analysis and extrapolation of full-scale engine noise with its multiplicity of sources is a difficult task. Furthermore, the near field data were acquired only 2.0 m from the jet axis, thus data interpretation is difficult.

Strout and Atencio (refs. 19–21) found good agreement between extrapolated near field data and far field data for a JT8D jet engine using the multiple-sideline source-location technique. However, their methods required manual fairing and extrapolation of the data at stages in the data processing, which required some knowledge of probable results gained from experience with jet noise. Our study involves an extension of Strout

and Atencio's techniques in which the data reduction is completely automated in a consistent manner and does not require manipulation of the data at various steps in the algorithm.

The S1 Wind Tunnel was used for this study because of its large speed range (0 to Mach 1) and large (8-m-diameter) test section. It was hoped that if the various difficulties with near field effects, background noise, and reflections could be dealt with, the advantages of high windspeed would lead to a better simulation of forward-speed effects than was possible in the lower-speed Ames 40- by 80-Foot Wind Tunnel (ref. 11). To cope with the anticipated acoustical problems, ONERA developed an elaborate test-section lining based on NASA's recommendation; that lining is described in this report. NASA provided the computer software developed for jet noise source location and extrapolation based on multiple-sideline noise measurements. (This software was subsequently improved by ONERA.) This report concentrates on the source-location technique and its results. The technique is based on the hypothesis that by mapping jet noise along two lines parallel to the jet it is possible, with certain manipulations of the data, to extrapolate to the jet and identify the apparent noise source regions at each frequency, as well as the radiation direction of the sources. Assumptions must be made about propagation decay and so-called near field effects. The noise can then be extrapolated to the far field. This report describes the techniques used, their advantages and disadvantages, and forward-speed effects on the jet and engine noise.

NOMENCLATURE

c	sound speed in ambient air, m/sec
d	ejector exhaust-nozzle diameter, 0.44 m
f	frequency, Hz
L _p	sound pressure level, dB re 2×10^{-5} N/m ²
M	jet relative Mach number, $0.62(V_j - V_a)/c$
M ₀	flight Mach number, V_a/c
R	distance from noise source location to observer, m
St ₀	Strouhal number (static case), fd/V_j
St	Strouhal number with wind, $fd/(V_j - V_a)$

T _t	jet total temperature at primary exhaust nozzle, °C
V _a	windspeed in test section or free stream, m/sec
V _e	effective jet exhaust speed, $V_j(1 - V_a/V_j)^{2/3}$, m/sec
V _j	jet exhaust speed at the primary nozzle exit (fig. 1), m/sec
X	distance along the jet centerline from exhaust nozzle to acoustic source, without wind, m
X'	distance along the jet centerline from exhaust nozzle to acoustic source, with wind, m
Y ₁	perpendicular distance from shear layer to near sideline, m (shear layer assumed to be at exhaust nozzle radius)
Y ₂	perpendicular distance from shear layer to far sideline, m
ΔL _p	difference or change in jet noise level, dB
ΔL _{p_i}	difference or change in internal engine noise level, dB
ΔL _{p_k}	kinematic effect on jet noise due to motion of airplane relative to observer, dB
α	normal incidence absorption coefficient
λ	acoustic wavelength, m
ψ ₁	angle between jet axis and line connecting exhaust nozzle center and observer on near sideline, with wind, deg
ψ ₂	angle between jet axis and line connecting exhaust nozzle center and observer on far sideline, with wind, deg
ψ _s	angle between jet axis and acoustic radiation vector connecting acoustic source and observer, with wind, deg
θ ₁	angle between jet axis and line connecting exhaust nozzle center and observer on near sideline, without wind, deg

θ_2	angle between jet axis and line connecting exhaust nozzle center and observer on far sideline, without wind, deg
θ_s	angle between jet axis and acoustic radiation vector connecting acoustic source and observer, without wind, deg
ρ_a	density of ambient air, kg/m ³
ρ_j	density of fully expanded jet, kg/m ³
Subscripts	
1	near sideline microphone traverse, or condition 1 flight and jet speed
2	far sideline microphone traverse, or condition 2 flight and jet speed

TEST EQUIPMENT AND METHODS

Jet Engine

The General Electric J-85 engine is a small turbojet with a maximum net thrust of 12,100 N, a maximum mass flow rate of 20 kg/sec, a maximum exhaust pressure ratio of 2.4, and an exhaust diameter of 0.44 m. The engine was operated with four nozzles: (1) a conical nozzle with ejector; (2) a variable flap ejector; (3) a 104-tube mixer suppressor nozzle; and (4) a 104-tube mixer suppressor nozzle with ejector. Only the conical nozzle with ejector made sufficient noise at all frequencies to dominate the wind tunnel background noise at the windspeeds used in this study; therefore data for the other nozzles will not be shown.

Figures 1(a)–1(c) show the engine geometry, including the convergent, primary exhaust nozzle and secondary cylindrical ejector. The cylindrical ejector had a blunt base, which was used in previous studies in which external pressure drag could be measured accurately (ref. 4). The ejector was designed to pump ambient air equal to only 5% of the primary flow for cooling purposes, and had no thrust augmentation; nor could the ejector be considered a mixing nozzle for enhanced jet decay and noise reduction. Moreover, the conical ejector nozzle diameter of 444 mm was used as the reference dimension for normalization of the distances used in the acoustic study. The primary exhaust nozzle was used as the reference for computation of jet thrust and velocity.

A flight inlet was used for both static and forward flight operation of the engine, because even in the static

case a low-speed airflow in the wind tunnel was induced by the engine.

Table 1 shows typical jet exhaust velocities, temperatures, momentum thrust, and pressure ratios used in this study. The jet was operated both subsonically and with sonic conditions at the primary exhaust nozzle. The jet pressure ratio was calculated from the ratio of total pressure in the jet to test-section static pressure. (The flow conditions at the primary nozzle were extrapolated from measurements made upstream (ref. 5).) At pressure ratios greater than 1.85, the flow at the primary exhaust reached sonic speeds. At higher pressure ratios, it is probable that a shock system existed downstream from the primary exhaust.

Although it was equipped with an afterburning duct, the engine was operated without afterburning.

S1 Wind Tunnel

The S1 Wind Tunnel at Modane-Avrieux (ref. 22) is a closed-circuit, continuous-operation, sonic wind tunnel with a test section speed range to Mach 1. Figure 2 is a schematic of the wind tunnel. The interchangeable, closed test section used in this study was 8 m in diameter and 14 m long. Two counterrotating fans are located in the crossleg downstream of the primary diffuser. There are no acoustic silencers in the wind tunnel circuit. The altitude of the wind tunnel is 1100 m.

Test Section Lining

The entire cylindrical test section was lined for this test with a 125-mm-thick absorbent lining composed of 75-mm polyurethane foam and a 50-mm air space as shown in figure 3. The outer 20 mm of foam was treated with chemical fire retardant. The foam was supported by a porous metal screen embedded 60 mm from the flow surface, and was backed by a perforated plastic film which improved the mid-frequency absorption. The air gap between the foam and the wall contributed to the low-frequency absorption illustrated in figure 4, which shows the acoustic absorption of the composite lining with 30- and 50-mm air gaps. The data were acquired with a standing-wave tube. The absorption was very good above 300 Hz. During the initial checkout in the wind tunnel, some of the panels started to vibrate badly when the windspeed reached 130 m/sec. An improved fastening technique solved this problem.

Model Installation

Figure 1(a) is a photograph of the J-85 engine and ceiling-mounted support strut in the test section. Figures 5(a) and 5(b) show the installation geometry. The center of the engine was 1.5 m above the center of, and on the vertical centerplane of, the test section.

Acoustic Instrumentation

Figures 5(a) and 5(b) illustrate the moving microphone traverse parallel to the jet axis. Two carriages were used, each equipped with two pairs of microphones. One microphone of each pair was the primary data microphone; the other was a backup. Four microphones (1, 2, 3, and 4) were traversed 2.0 m from the jet centerline, and four microphones (5, 6, 7, and 8) were traversed 3.5 m from the jet centerline. The two rails were in line such that the upstream carriage with microphones 1, 2, 5, and 6 traversed 6.7 m from the J-85 inlet to a point just downstream of the exhaust nozzle, and the second carriage with microphones 3, 4, 7, and 8 continued along the jet exhaust another 6.7 m. This arrangement allowed a sweep of the jet noise radiation relative to the exhaust center of 40° to 165° at the 2.0-m sideline and 56° to 155° at the 3.5-m sideline, 0° being the upstream direction. The microphone carriage and rail were streamlined and curved for minimum acoustic reflection, but they were not acoustically treated. The microphones with nose cones were pointed upstream and were essentially omnidirectional. The carriages moved simultaneously and took approximately three minutes to complete the traverse. An optical device tracked the carriage position to the nearest millimeter.

The guy wires shown in figure 5(b) were used to secure the microphone arrays. They generated a tone near 2.5 kHz, but the tone is not visible in the third-octave band spectra.

Also shown in figure 5(a)–5(c) is the fixed array of 48 microphones, 3.2 m from the jet axis, used by ONERA to locate acoustic sources by means of antenna signal processing. In addition, eight of the microphones with 15-cm spacing and eight microphones with 30-cm spacing were connected to an electronic time-delay system developed at Ames to create two eight-channel broadside antennas (ref. 23). A broadside antenna parallel to a jet can be focused on a region of the jet and reject sound arriving from upstream or downstream, to varying degrees, depending on the number of array elements. The antenna cannot, however, discriminate in the vertical plane to reject floor or ceiling reflections. The antennas were focused on various parts of the jet exhaust for on-line analysis of the

jet noise. (The data for this analysis are not reported here.) The other data systems were processed off line.

The moving microphone data were recorded on analog recorders with appropriate time-code signals so that analysis in third-octave bands could be made of the sound as a function of position. Amplifier gain for each channel was set automatically and then locked just before a traverse. Gain was recorded on each channel using a frequency code. Test number, run number, date, microphone number, and wind tunnel identification were recorded on each data channel as a pulse code interpreted as binary numbers. The third-octave band data were digitized after the test and input to the source-location/extrapolation computer program to be described.

Engine Instrumentation

Knowledge of the engine performance was required to evaluate the acoustic data. The J-85 was instrumented for (1) total pressures upstream and downstream of the compressor, (2) temperatures downstream of the compressor, (3) fuel flow rate, and (4) engine rotational speed. The instruments provided fluid mechanics data sufficient for the computation of thrust, mass-flow rate, and jet exhaust velocity. The instrumentation is described in reference 5.

Test Procedure and Limits of Variables

The first data sets were obtained during static operation of the engine in the wind tunnel. Because of temperature limitations of certain engine components, the maximum engine speed attained was 16,170 rpm (98% of the maximum rated rpm), which produced a maximum jet exit velocity of 545 m/sec during static operation and 606 m/sec at forward speed. Operation of the engine induced a relatively small but significant airflow in the wind tunnel of 16 m/sec maximum. Acoustic data were taken at several windspeeds, up to a maximum of 130 m/sec. The windspeed was limited by the allowable loads on the acoustic lining. Nonetheless, the maximum speed was significantly greater than the maximum speed of 91 m/sec previously attained in the Ames 40- by 80-Foot Wind Tunnel. (Since these tests, the maximum speed in the 40- by 80-Foot Wind Tunnel has been increased to 155 m/sec.)

The general procedure was to set the wind tunnel and engine speed, take data while the microphones were traversing downstream, and then increase the engine speed while moving the microphones back to their starting positions upstream. Background noise was measured with the engine in the test section but not operating.

Data Analysis

Extrapolation to the far field of noise levels measured close to a large distribution of noise sources such as a jet exhaust would be hopeless without some knowledge of the source regions and their directional properties as functions of frequency. Different researchers have employed a variety of experimental techniques for identifying the source regions of jets, including use of acoustic antennas, focusing reflectors, cross correlation, infrared detectors, and in-flow probes. The multiple-sideline source-location technique (ref. 16) was developed on the premise that a proper map of the acoustic field at two distances from a jet contains enough information to describe how and from where the sound is propagating. Once this information is known, a single map of the sound field can be used to extrapolate the jet noise to any distance. To apply the method to this project, the following steps were taken.

1. The jet noise was recorded along two lines parallel to the jet axis, at locations 2.0 m and 3.5 m from the jet axis, as shown in figures 5 and 6. The traverse lines and the jet were nearly in the same plane. The 2.0-m and 3.5-m sideline data from the S1 Wind Tunnel study could have been used for determination of source locations had it not been for a small acoustic interference resulting, we suspect, from low-frequency propagation around the circuit. The acoustic lining did not absorb well at low frequencies, and low-frequency sound propagates well in wind tunnels. Consequently, low-frequency noise measured upstream of the jet was louder than it should have been. Because of this problem, noise data from the same engine operated statically at the Ames outdoor test site (ref. 2), recorded at 2.0-m and 12.0-m sidelines (measured from the jet centerline), were used for the source-location part of the analysis. The data were acquired with slowly traversing microphones. Details of the engine and microphone data acquisition are given in reference 2.

Despite the small amplification in the S1 Wind Tunnel, the S1 data and the Ames outdoor data agree fairly well, as illustrated in figure 7, which shows overall sound levels measured on a 2-m sideline at ONERA and at Ames. The figure also shows the Ames data corrected to the same jet speed and forward velocity condition (from wind-tunnel flow induced by the engine) as in the S1, using the method of reference 24. (This method will be discussed further in the Results and Discussion section.) The S1 data rise above the Ames data at small radiation angles, to a maximum of 3 dB at 40°. This same trend was seen in data obtained by other experimenters. At large angles, the jet was strong and it dominated most contaminations. At small radiation angles, upstream of the exhaust, the low-level sound at low frequencies was con-

taminated by the jet noise which radiated downstream and traveled around the circuit. In any case, this effect would be generally similar with wind or without wind, so that the measured *change* in jet noise caused by flight (i.e., flight noise minus static noise) should be relatively unaffected.

The difference between the S1 and Ames data shown in figure 7 was subtracted from certain S1 directivity plots (figs. 21–24) (see Results and Discussion). This correction ranged from 0 dB at $\psi_2 = 135^\circ$ to 3 dB at $\psi_2 = 40^\circ$ for overall sound levels. Similar comparisons in third-octave bands showed somewhat larger differences at low frequencies and somewhat smaller differences at high frequencies (see table 2). The source of the high-frequency amplification in the S1 is unknown. The third-octave band plots in figures 21(a)–2(f) were corrected for the amplifications listed in table 2.

2. The acoustic data were averaged and converted to third-octave band spectra at specific angles relative to the engine exhaust. Because the microphones were continuously moving, the maximum averaging time was 2 sec for a particular angle, leaving some scatter in the spectra (e.g., third-octave analysis of stationary Gaussian noise at 250-Hz and 2-sec integration time results in ± 1 dB accuracy for 99% confidence; scatter is less for higher frequencies, more for lower frequencies or nonstationary noise). The data were also corrected for microphone frequency response by increasing the measured sound level where the response was low, and vice versa. Next, the spectra were smoothed by fitting an 8th-degree polynomial to the data. Figure 8 shows typical data at the 2.0-m sideline before and after the curve fit.

3. The acoustic spectra were then replotted as third-octave band levels versus exhaust-microphone angle at each frequency. It was again necessary to smooth the curves with an 8th-degree polynomial, because the source-location method requires well defined peaks in the noise-directivity plots. Any anomalous peaks resulting from data scatter complicated the extrapolation, especially since the operation was done automatically by the computer.

From static data acquired at Ames (ref. 2), such as those plotted in figures 9(a) and 9(b), pairs of angles were found which define the propagation direction at each frequency. That is, it is assumed that the far field peak in the directivity plot was generated by the same acoustic ray that passed through the near field peak. Similarly, it is assumed that each segment of the far field plot is related to a particular segment of the near field plot by the same noise difference that was found for the peaks. This is

illustrated graphically in figures 9(a) and 9(b). Thus, for each third-octave frequency band and each far field angle there is a corresponding near field angle through which the sound ray at that frequency passes. In other words, a sound ray propagating from the jet would pass through the near and far sidelines at two points along that ray which can be described by a specific pair of angles θ_1 and θ_2 .

This procedure can break down, however, if the far field and near field curves tend to converge, which sometimes happens at high frequencies and low directivity angles as shown in figure 9(b). In this case, one cannot find an appropriate noise difference at low angles equal to peak noise difference, as one can in figure 9(a). The anomalous rise in the far field curve (fig. 9(b)) would cause the computer to find a noise difference for an illogical angle pair, which would result in an incorrect source location. It would be tempting to repair the lower curve of figure 9(b) at low angles to force the angle pairs to some "appropriate" values, but this adjustment is not justified by the data and was not done.

Figure 10 illustrates the peak noise radiation angle plotted versus Strouhal number as measured at Ames and in the S1 Wind Tunnel. There was reasonably good agreement between the Ames and S1 data at similar jet exhaust speeds. As expected, there was a tendency for the higher frequency sound to radiate at smaller angles than the low frequency sound did.

4. Tracing the acoustic ray at each frequency back to the jet defines static source locations shown, for example, in figures 11(a) and 11(b) for two Strouhal numbers. Figures 11(a) and 11(b) are from the Ames static test of reference 2. It was assumed that the sources were located in the near shear layer, one nozzle radius from the centerline. This arbitrary definition of a source center is an obvious simplification of a complicated phenomenon because it says nothing about source distribution or multiple locations in the jet radiating noise to the same point in space.

The source-location method does not require that all sound at a given frequency must originate from the same point in the jet. The data show that a range of source locations at one Strouhal number (or frequency) radiate sound at different angles. Note that in most cases, lower frequency sound and sound radiated at large angles tend to originate farther downstream than high-frequency or low-angle sound do, which is consistent with results in reference 16. Figures 12(a) and 12(b) are a comparison of source locations measured at Ames and in the S1 Wind Tunnel for the static case. At radiation angles $\theta_s > 60^\circ$ and frequencies above 500 Hz, the agreement is reasonably

good. At low frequencies and low radiation angles, the suspected S1 acoustic contamination distorted the computation of the source locations. For this reason, the Ames data were used for this phase of the data analysis.

5. Once the source locations and radiation angles for a given frequency or Strouhal number are found, the data measured anywhere in the field, usually in the near field, can then be extrapolated to any far field distance. This extrapolation, however, is the weakest part of the method because, as experience has shown, the near field noise levels are always less than one would get by measuring in the far field and extrapolating back to the near field point using spherical radiation (6 dB per double distance plus correction for atmospheric effects). This deficiency may be a large-scale-jet effect; Ahuja et al. (ref. 15) reported no near field effect for a small-scale jet (2.54-cm-diameter nozzle) when the source locations and noise directivities were identified. However, the near field effect is consistent with the concept of distributed or finite line sources with sound fields that decay at 3 dB per double distance out to a particular radius and 6 dB per double distance beyond that, neglecting atmospheric absorption.

Without information on the extent of the distributed sources, the decay rate cannot be known even if the source center is known. To extrapolate the sound to a specific point, one must develop a so-called near-field correction. This near field correction is simply the difference at the near sideline between the measured levels and the levels extrapolated from a far field sideline using spherical decay and atmospheric absorption. The correction requires the operation of the engine outside the wind tunnel to get far field data.

Figure 13 illustrates typical near field correction curves measured with the conical nozzle and various jet velocities acquired at Ames in the study described in reference 2. The data collapse to a single curve when plotted versus the nondimensional parameter $(R/\lambda)(V_j/c)$, in which R is the distance to the near field measurement point. The correction should change with nozzle type. It is not known if these near field corrections found statically hold true for the jet in flight. Since the jet core probably stretches in flight, the noise source distributions possibly change enough to affect the near field correction.

6. The source location, X , and radiation angles, θ_s , acquired statically must be corrected for convection effects in the wind tunnel (ref. 2), as illustrated in figure 14. The notations for the static parameters X , θ_s , θ_1 , and θ_2 become, with wind, X' , ψ_s , ψ_1 , and ψ_2 , respectively. It is assumed that the wind causes the jet core to

stretch, moving the noise sources downstream. This effect is handled by redefining the Strouhal number of each source location plot so that (ref. 16)

$$St = fd / (V_j - V_a) \quad (1)$$

Equation (1) is somewhat arbitrary in that it implies that the movement of noise sources downstream in the jet (or stretch) because of ambient wind is equivalent to increasing the source frequency (or Strouhal number) at any fixed point in a coordinate system fixed to the engine. This is probably based on evidence that high-frequency sources are closer to jet exhaust nozzles than low-frequency sources are. Hence, at a given location in the jet, increasing forward speed will cause high-frequency sources to move downstream and to displace the lower frequency source that was there. Equation (1), therefore, is an estimate of how those source frequencies (or Strouhal numbers) change.

In the source-location method, it is assumed that as wind increases, the static plots of source location versus radiation angle are perfectly valid for wind-on conditions, at the appropriate Strouhal numbers given by equation (1). Although this reasoning agrees with experimental trends, the exact relationships between forward speed, source movement, source radiation, and source Strouhal number have not been verified except by the comparisons between the final extrapolated noise field and far field data (as will be discussed). Furthermore, these source convection effects are valid only for pure jet noise sources and are not valid for other sources fixed to the engine such as combustion noise, turbomachinery noise, or (possibly) shock noise.

In the wind tunnel, sound waves generated during simulated flight are swept downstream and the new source radiation angle is defined as (ref. 13)

$$\psi_s = \tan^{-1} \left\{ \left[\sin(\theta_s - 90^\circ) + V_a / c \right] / \left[\cos(\theta_s - 90^\circ) \right] \right\} + 90^\circ \quad (2)$$

Convection of sound waves in a wind tunnel relative to the engine is equivalent, with respect to direction, to the movement of the aircraft in flight away from the propagating sound waves. So, for a coordinate system fixed to the aircraft, the acoustic field directivity pattern is the same in both cases.

Figures 15(a)–15(d) show jet noise source locations with wind that were deduced from static source locations

by means of the above equations. Those results indicate that, at a given Strouhal number, a range of source locations radiate sound over a range of angles.

7. With the above information, it was then possible to correct the S1 Wind Tunnel data for near field effects, and then extrapolate to far field. For a given frequency and location along the 2.0-m sideline, the convected source locations and radiation angles were examined until a source location and radiation angle were found that corresponded to an acoustic ray propagating through the measurement point. The corrected wind-tunnel data were then extrapolated to the desired far field sideline along that ray using spherical decay and atmospheric decay. Table 3 shows typical data, corrections, and extrapolations. Figures 16(a) and 16(b) illustrate typical 2.0-m data and extrapolated results at the 122-m sideline. The accuracy of the extrapolation is very sensitive to the accuracy of the source location. A source position error equal to one nozzle diameter can lead to a far field radiation angle (θ_s or ψ_s) error of up to 12° for that source.

The computer code developed to perform all the corrections, curve fits, calculations, extrapolations, and plotting is complex. Figure 17 illustrates the flow of information required to carry out the numerous operations automatically. Reference 25 describes the function of each module of the code as it was developed to operate on the ONERA computer systems. Appendixes A and B contain the computer code listings for the main programs, NOISE3 and NOISE4. NOISE3 calculates the source location with the wind off, and NOISE4 calculates the far field directivity and wind effects.

RESULTS AND DISCUSSION

Background Noise

Figures 18(a)–18(c) show typical background noise levels measured at the 2.0-m sideline in the S1 Wind Tunnel for windspeeds of 87, 113, and 130 m/sec. The data are presented as sound pressure level in third-octave bands versus angle relative to the nozzle exit center (0° is upstream, 180° is downstream). This is the typical format for this report. Sound pressure level peaks at around 114 dB. The noise levels appear to be dominated by flow interaction with the microphones and support fairings. These levels are sufficiently below the jet noise levels as to be negligible. Since the 3.5-m-sideline jet noise data were contaminated by background noise at high windspeeds, none of the 3.5-m data are presented in this report.

Extrapolation to Far Field

Figures 19(a)–19(j) show the 2.0-m-sideline jet noise levels in the S1 Wind Tunnel and the corresponding extrapolated noise at the 122-m sideline for windspeeds of 49, 72, 88, 113, and 130 m/sec. The 2.0-m data show peak noise levels at 400–630 Hz and peak radiation angles of 155°–160°. At 122 m the noise peaks at 400–500 Hz and radiation angles of 142°–160°, which is appropriate for source locations about two nozzle diameters downstream of the exhaust (see fig. 15(b)). In other words, a sound ray originating in the shear layer two nozzle diameters downstream of the exhaust and passing through the 2.0-m sideline with an angle, ψ_1 , of 155°–160° will arrive at a 122-m sideline with an angle, ψ_2 , of 149°–156°, according to the geometry (see fig. 14).

The data in figures 19(a)–19(j) are plotted directly from the automatic data-reduction program and have obvious discontinuities which appear at some high frequencies and high angles. The discontinuities are the result of rapid, fictitious changes in apparent source locations. For example, the discontinuity in the 8-kHz curve in figure 19(f) can be traced to the scattered source locations at 8 kHz in figure 15(d). These abnormal source locations are created by anomalies in the spectra used to find the angle pairs (figure 9(b), for example), which the computer code treats impartially. However, the resulting error in the extrapolations of figure 19 have a negligible effect on the overall noise levels.

Internal Engine or Shock Noise

It is important to note that at the higher jet speeds, there is evidence that internal engine noise or broadband shock noise is dominating the near field jet noise at low directivity angles and midfrequencies, but it is not clear which of the two noise sources is dominant. Figures 19(g) and 19(i), for example, show peaks, or lobes, in the near field directivity pattern between 60° and 120° that are obviously not caused by pure jet noise. When these noise lobes are extrapolated to far field, only the lobe at around 120° is clearly visible (figs. 19(h) and 19(j)).

The changes in noise with jet speed for 1.0 kHz and 1.2 kHz are plotted in figure 20, which illustrates the rapid growth of internal or shock noise in the near field and the slower growth in the far field. Although the data set is limited, the curves indicate that in this operating regime, the near field or shock noise increased with jet speed (neglecting differences in forward speed), as

$$\Delta Lp_i = 10 \log(V_{j2} / V_{j1})^{24} \quad (3)$$

and the far field internal and jet noise increased, as

$$\Delta Lp = 10 \log(V_{j2} / V_{j1})^5 \quad (4)$$

This apparent contradiction in growth rates can only be possible if the near field microphone was dominated by a nearby internal or shock noise source when the microphone passed upstream of the exhaust nozzle. In that region, the jet turbulence noise is relatively weak and the internal or shock source strength grows rapidly with jet speed. In the far field, however, the microphone is about equidistant from sources throughout the jet, and the radiation angle is larger so that jet noise is strong, yet growing at a slower rate than internal noise. As jet speed increases, however, the proportion of far field noise caused by internal or shock sources must increase. When one type of source dominates both the near and far acoustic fields, the near field and far field growth rates with increase in jet speed will be identical.

Thus, the J-85 far field directivity pattern at midfrequencies tends to have two major lobes, one near 160°, caused by pure jet noise, and one near 120°, which appears to be influenced by internal or shock noise. This latter observation is supported by Stone's plot in reference 26 of internally generated noise directivity from many full-size jet engines. This plot shows a broad peak that reaches 120° and has a shape very similar to the mid-to high-frequency data plotted in figure 19. Internally generated noise could also be important at angles lower than 120°, where the jet noise is relatively weak. These two lobes are not clearly seen in the overall sound-level directivity plots.

Forward Speed Effects

We can now compare the data of figures 16 and 19 to show the effects of forward speed on jet noise. Figures 21(a)–21(e) show the flight effects on the overall sound pressure levels at the 122-m sideline for speeds of 49, 72, 88, 113, and 130 m/sec.

Corrections

All curves in figures 21(a)–21(e) were corrected equally for the 0–3-dB amplification in the S1 Wind Tunnel described in the data analysis section. Because the static data were recorded with an induced windspeed of 16 m/sec in the wind tunnel and a jet speed of 545 m/sec, it was also necessary to correct the far field static data to zero windspeed and to the correct flight jet speeds noted in figures 21(a)–21(e). Both original and corrected curves are shown. These corrections were taken from the

semi-empirical jet noise prediction of Stone et al. in reference 24, and ranged from less than 1 dB in figure 21(a) to a maximum of 6.6 dB at 140° in figure 21(e). The change in jet noise due to a change in jet and/or flight speed is given by

$$\Delta Lp = Lp_2 - Lp_1 = 10 \log(\rho_{j2} / \rho_a)^{w_2} - 10 \log(\rho_{j1} / \rho_a)^{w_1} + 10 \log(V_{e2} / V_{e1})^{7.5} - 15 \log(K_2 / K_1) \quad (5)$$

where ΔLp is the change in noise for any change in flight speed or jet speed, represented by "condition 1" (Lp_1) or "condition 2" (Lp_2). For each condition, the following parameters are computed.

$$V_e = V_j(1 - V_a V_j)^{2/3} \quad (6)$$

$$w = 3(V_e c)^{3.5} / [0.6 + (V_e / c)^{3.5}] - 1 \quad (7)$$

$$K = [1 + M \cos(\psi_2)]^2 + 0.04 M^2 \quad (8)$$

$$M = 0.62(V_j - V_a) / c \quad (9)$$

Equation (5) gives the change in overall noise due to the change in jet speed and flight speed of a simple isolated jet parallel to the flight direction. Internal engine or shock noise sources are not included. The dynamic effect on sound amplitude of the change in relative motion between the source and the propagation medium is included, but the kinematic effect of relative motion between aircraft and observer is not included since there is no such motion in the wind tunnel. The kinematic effect is small; an estimation of it is given in the section "Comparisons with Predictions". It was deduced from the results of reference 24 that, to a first approximation, equation (5) is also valid for third-octave band frequencies greater than or equal to 500 Hz. However, ΔLp should be reduced by 25% at 250 Hz and by 50% at 125 Hz, according to reference 24.

Because the high-speed jet noise in figures 21(d) and 21(e) was affected by internal or shock noise as discussed above, corrections based entirely on equation (5) would be inappropriate in that case. So, for angles equal to or less than 90°, equation (4) was used to correct the mix of internal and jet noise in the forward quadrant. For larger directivity angles where jet noise dominates, equation (5) was used for the corrections. As before, the magnitude of

the correction is indicated by the difference between the dashed line and the circles in figures 21(d) and 21(e). This procedure does not entirely resolve the uncertainty inherent in figures 21(d) and 21(e), which show comparisons of static (nearly pure) jet noise with forward speed jet/internal noise. Because of engine temperature limitations, the static data were acquired at jet speeds low enough that internal noise was not evident. Thus, the static and forward speed noise sources are not identical despite the attempt to correct the static data for equivalent conditions.

Discussion

At a forward speed of 49 m/sec (fig. 21(a)), the simulated flight noise is less than the static noise at large radiation angles, as would be expected from the smaller relative velocity between the jet and the ambient air. At low radiation angles, the flight effect on jet noise decreases to around zero at 50°, as reported by many researchers. Stone (ref. 26) attributes this effect primarily to internally generated noise which dominates as jet noise decreases with forward speed. Internally generated noise can arise from many sources, such as combustion and turbomachinery. As forward speed increases, the peak flight noise near 140° decreased relative to the static case by a maximum of 10.5 dB at a forward speed of 130 m/sec (fig. 21(e)). In all cases, the flight and static data tend to converge as the radiation angle decreases to 40°. Amplification or crossover of jet noise due to flight was noted at small angles in figures 21(a), 21(d), and 21(e).

Equation (5) indicates that, in flight, pure jet noise should decrease in the forward quadrant (low directivity angles) relative to the static case. On the other hand, internal noise should increase in the forward quadrant because of source motion relative to the medium (dynamic effect) and source motion relative to the observer (kinematic effect). Following Stone's reasoning (ref. 26), the dynamic effect on internal noise relative to the static case can be estimated by

$$\Delta Lp_i = -30 \log[1 - M_o \cos(\psi_2)] \quad (10)$$

(Stone uses a -40 multiplication factor because he lumps the dynamic and kinematic effects, but the latter is not present in the wind tunnel). Thus, if the data in figures 21(a)-21(e) had been generated by pure jet sources, the separation of the curves in the forward quadrant would be greater, according to theory. Conversely, if both sets of data had contained strong internal noise, the curves would cross in the forward quadrant, which they do slightly in some cases. This is an example of the

difficulties of working with jet engines rather than pneumatic nozzles. Full-scale jet engines have jet and internal noise sources that are difficult to distinguish in the far field except with forward speed, when the two types of noises react differently.

The attenuation of J-85 aft-quadrant noise ($\psi_2 > 90^\circ$) in flight and amplification of the forward-quadrant noise ($\psi_2 < 90^\circ$) in flight simulations using the Bertin Aerotrain were reported by Drevet et al. (ref. 8). (The Bertin Aerotrain is a French air-cushion vehicle that moves along an inverted "T" track.) The J-85 was modified with an inlet duct lining to suppress compressor noise that might have affected jet noise measurements in the forward direction. At a flight speed of 82 m/sec and a jet speed of 505 m/sec, there was an aft-quadrant attenuation of 8 dB and a forward-quadrant amplification of 1 dB. Both static and flight directivity patterns peaking near 140° were shown. For the S1 data in figure 21(c), the attenuation of peak noise levels due to forward speed is 6 dB and the attenuation at 40° is less than 1 dB. These numbers go to 7 dB and 0 dB if the wind tunnel data are corrected for kinematic effects present in the Aerotrain test, as will be explained in a later section. Hence, the two data sets are in reasonably good agreement with regard to peak noise attenuation.

From the Aerotrain data of reference 8, Hoch (ref. 7) proposes that forward speed effects on jet noise can be explained using the concept of shear noise and self noise in the jet. The aft-quadrant noise is dominated by jet shear which becomes weaker as the relative speed between the jet and the ambient flow decreases. The forward-quadrant noise, according to Hoch, is dominated by excited turbulence or self noise in the jet interior which is less affected by the external flow speed. Hence, the aft-quadrant noise decreases with forward speed, but the forward-quadrant noise does not. The results of the present study cannot disprove Hoch's ideas of internal jet turbulence noise with respect to pure jets, but the data (e.g., figs. 19(g) and 19(i)) support Stone's arguments about internal sources dominating forward-quadrant noise in flight. The forward-quadrant noise in figures 19(g) and 19(i) looks quite unlike jet or turbulence noise, as if a new source were developing, such as might be produced by turbomachinery, combustion, or shocks.

The third-octave-band noise levels, wind on, are compared to the zero wind data in figures 22(a)–22(i), which illustrate 122-m-sideline data at zero and 130 m/sec windspeeds for third-octave bands at 125, 250, 500, and 800 Hz and 1.25, 2.5, 5.0 and 8.0 kHz. As before, both curves were corrected for S1 amplifications as listed in table 2, and the static data were corrected to the appropri-

ate wind and jet speeds using equation (4) for $\psi_2 < 90^\circ$ and equation (5) for $\psi_2 \geq 90^\circ$. The static data are plotted with and without the wind/jet correction. The low-frequency data (125 and 250 Hz) show a reduction of jet noise due to forward speed at all angles. At other frequencies, crossover of forward-quadrant noise occurred. The 500- and 800-Hz data dominate the overall sound levels (see fig. 19(j)), which explains why the overall sound levels converged at small angles.

Comparisons with Published Data

In figures 23(a)–23(c), comparisons are shown between the S1 Wind Tunnel results and J-85 data acquired in other flight tests or simulations with respect to overall sound levels. The curves labeled Aerotrain were taken from reference 6, which describes a joint General Electric/SNECMA test of a J-85 engine on the Bertin Aerotrain. The curves labeled Learjet are also from reference 6 and represent tests of the Gates Learjet powered by two J-85 engines. All three sets of data in figures 23(a)–23(c) correspond to noise on a 122-m sideline. In all cases, the flight and train data were corrected to a common forward speed and jet velocity using equation (5) plus the following correction for kinematic effect of the relative motion between the source and the observer (this effect is not present in wind tunnel data) from reference 24.

$$\Delta L_{p_k} = -10 \log(1 - M_{o2} \cos \psi_2) + 10 \log(1 - M_{o1} \cos \psi_2) \quad (11)$$

The agreement among the data is fair, considering the differences in the experimental methods and the potential errors inherent in each. For example, the Aerotrain J-85 had treated inlet ducts, and the Learjet J-85 noise directivity was affected by wing and fuselage reflections. Furthermore, the accuracy of flyover noise tests are often poor due to variations in source position, short sampling times, ground reflections, and atmospheric effects. At angles of less than 90° , the S1 Wind Tunnel data are consistently higher than the Aerotrain data. Since internal noise plays an important role in J-85 forward-quadrant noise, as previously discussed, the Aerotrain duct treatment may have suppressed that noise to some extent. From 90° to 140° , the S1 and Aerotrain data agree fairly well. The largest differences are for angles above 140° . The S1 peak levels are around 2 dB higher and at a greater angle by about 5° than the Aerotrain peak levels. Considering the variety of experimental methods and corrections employed, the data agreement is reasonable.

Figure 24 is a comparison of overall sound levels of the J-85 engine measured in the S1 Wind Tunnel with those measured by Atencio (ref. 11) in the Ames 40- by 80-Foot Wind Tunnel before the present acoustic wall lining was installed (ref. 27). There were important differences between the two studies. The Ames data were measured 4.3 m directly below the engine, which was mounted below an aircraft model wing; the engine was 6.1 m above the hard floor. The data were extrapolated to 30.5-m flyover distance by means of an early version of the source-location technique. Furthermore, the Ames data contains corrections for wind tunnel reverberations and near field effects. Because the 40 by 80 test section was unlined, the reverberation corrections intended to correct the data to free field conditions are quite large, ranging from -9 to +12 dB. The 40 by 80 data as published also contain a wind correction factor so that the results simulate static noise propagation. This correction was removed for the comparison in figure 24 so that the S1 and 40 by 80 wind effects are similar. The S1 far field noise levels were extrapolated from 122-m sideline to 30.5-m sideline by means of spherical radiation. The agreement is difficult to evaluate in detail because of the limited number of data points in the 40 by 80 data, and because of possible errors in the large corrections of the Ames data. It appears that the S1 sound levels are about 1-5 dB higher than the 40 by 80 levels. The 40 by 80 data appear to lack the peak noise.

Comparisons with Predictions

By plotting the difference of jet noise measured in simulated flight and that measured statically, $L_{p\text{flight}} - L_{p\text{static}}$, one can compare the experimental results with the flight effect predictions given by equations (5) and (10). Flight effects on jet mixing noise (eq. (5)) account for two factors: (1) source strength alteration resulting from the external flow around the jet plume, and (2) the dynamic effect of the relative velocity between the jet and the ambient air. Because there is no relative motion between the airplane and the microphone in a wind tunnel, the kinematic effect is not present in wind tunnel data, nor in equation (5). Since internal engine noise is an important component of J-85 noise at small angles, the flight effects on internal noise can also be compared with the experimental results.

Predicted flight effects from equation (5) (jet noise) and equation (10) (internal noise) were compared to the data of figures 21(a)-21(e) as shown in figures 25(a)-25(e). The difference between flight and static noise, measured and predicted, are plotted. The curves all have similar trends (slope) with respect to direction. The levels, however, differ. At directivity angles of less than 90°, the

measured change in noise due to forward speed falls between that predicted for internal noise, which is positive, and for jet mixing noise, which is negative. Since the near field data indicate that the J-85 generates a mixture of internal and jet noise in the forward quadrant, these results are reasonable. At lower jet speeds, where internal noise is not strongly evident, the measured results ($\psi_2 < 150^\circ$) are within 2 dB of the prediction for pure jet noise (eq. (5)). At jet speeds greater than 577 m/sec, where the near field data indicate strong levels of internal noise ($\psi_2 \leq 90^\circ$), the measured flight effects are closer to the predictions of equation (10) for internal-noise flight effects. Between 90° and 140°, the measured and predicted jet-noise flight effects (eq. (5)) agree reasonably well. This is also consistent with fact that jet noise dominates internal noise in the aft quadrant. Beyond 140° there is a discrepancy between measured and predicted flight effects. This is evident in figure 15, which shows that the source locations change rapidly in the negative direction above 140°; that is, they appear to disappear into the engine, and are poorly defined. Thus, the source location method fails at high radiation angles.

To summarize, the prediction of the change in pure jet noise due to forward speed given by equation (5) agrees within 2 dB of measured flight effects for jet speeds equal to or below 577 m/sec (excluding the faulty data above 140°). The data are consistently weaker (closer to zero) in the forward quadrant and stronger in the aft quadrant than predicted by equation (5). At higher jet speeds, the data in the forward quadrant approach, but do not reach, the internal noise prediction of equation (10). For this type of jet engine, a reasonable approach for predicting flight effects would be to split the difference between equations (5) and (10) for the forward quadrant, and to use equation (5) for the aft quadrant.

No prediction of absolute noise levels of the J-85 is presented, because that would require information on the engine internal noise sources not available in the literature. Attempts to use the pure jet prediction method of Stone et al. (ref. 24) proved unsuccessful because the internal sources are important on the J-85.

Figures 26(a) and 26(b) compare the S1 and Aero-train data (ref. 6) with the flight effect predictions at forward speeds of 41 and 82 m/sec, respectively. Three prediction curves are shown—one for internal noise only (eq. (10)), one for jet noise without kinematic effect (eq. (5)), and one for jet noise with kinematic effect added (eq. (5) plus eq. (11)). This last prediction is incorporated because the Aero-train data should show a kinematic effect of relative motion between the source and the receiver, whereas the S1 data should not. This results in a small

predicted amplification of pure jet noise caused by the motion between the aircraft and observer (the kinematic effect), as illustrated. Again, all curves have similar slopes. Unfortunately, the two data sets are not consistent enough to confirm the existence of the kinematic effect. As before, the S1 data fall between the predicted internal and jet noise curves at most angles because of the mixture of jet and internal noise in the data. The S1 and Aerotrain data agree within a few dB, out to 140° . Above that angle the S1 data are in error because of the shortcoming of the source location/extrapolation method. Thus, the total S1 flight-effect results, which include source strength alteration and the dynamic effects of jet/ambient relative velocity, have the proper trend with regard to flight velocity and radiation angle, according to theory and to comparison with Aerotrain data, except at angles above 140° .

CONCLUDING REMARKS

A study of flight effects on noise from a J-85 turbojet engine with a conical, convergent nozzle tested in the ONERA S1 Wind Tunnel has confirmed the strong effect of forward speed on noise from a full-scale engine. At the top simulated flight speed of 130 m/sec, the peak overall noise levels in the aft quadrant were attenuated approximately 10 dB relative to noise generated statically, and the forward quadrant noise increased slightly at 40° directivity angle.

The data indicated that internal engine and broadband shock noise make an important contribution to J-85 noise, particularly at high jet speeds, but it was unclear which of the two sources was dominant. The internal and shock noise components vary with changes in jet speed and forward speed in a different manner than does pure jet noise. A new empirical equation was presented that relates changes in internal or shock noise to changes in jet speed. The data were also compared with Stone's jet noise predictions. The comparison indicates that, in general, J-85 flight effects can be predicted to within 2 dB using equations for pure jets, except for the case of forward-quadrant noise ($\psi < 90^\circ$) from high-speed exhausts. In that case, estimates of internal-noise flight effects must be incorporated in the prediction. Kinematic amplification caused by aircraft motion relative to the observer (not present in wind tunnel data) could not be confirmed by comparing these results with other published data.

The S1 data were also compared with published data from studies of J-85 noise in an outdoor static test, a flight test, a wind tunnel test, and a moving train test. In general, the trends in the data were reasonably similar, considering the variety of test techniques used. Nonetheless, there were anomalies in the S1 data. At low frequencies and

low angles (forward quadrant), there was a 0- to 3-dB contamination of S1 data as a result of sound propagation around the circuit. This effect was quantified by comparing S1 data with free-field data to arrive at suitable correction factors. At directivity angles greater than 140° , however, the data deviated greatly from predictions and from other data sets because of the breakdown of the source-location/extrapolation technique. This breakdown might have been alleviated if flow noise had not prevented the use of sideline data measured 3.5 m from the jet axis. All results of this study are based on acoustic data acquired 2.0 m from the jet, which is a sideline distance that necessitated a substantial correction for near field propagation effects.

One of the objectives of the program was to automate the Ames version of the multiple-sideline source-location method to create an efficient, operational data-reduction system for jet noise studies that would not require intermediate manipulation of the data during processing. That objective was not achieved. The data reduction was automated, but the computer codes were complex and were so difficult to debug that the data-reduction system was not efficient. The only way the output could be checked was by comparing the final results with other published data, as discussed in this report. It could not be determined whether the anomalies in the results were due to errors in the computer code, or to the experimental method used, or to physical uncertainties related to the problem of measuring sound in the near field of distributed sources. Furthermore, for the near field effect, a correction factor had to be found during static operation of the engine and applied to flight data—a technique that has not been verified.

These experimental difficulties could be alleviated if large-scale jet noise studies such as this were done in the recently modified 40- by 80-Foot Wind Tunnel. The large test section, acoustic wall treatment, long circuit, and acoustically treated corner vane set (refs. 28 and 29) would allow collection of acoustic data of sufficient quality that it would not be necessary to push the multiple-sideline technique to its limits. At high speeds, background noise in the 40×80 is only a few dB quieter than in the S1, because the high-speed noise in both facilities is dominated by flow noise in the test section. However, acoustic antennas could reduce the effective background noise (ref. 23) and permit sideline traverses in the 40×80 that are farther from the jet than is possible in the S1. The problems encountered in the S1 with near field effects and noise propagation around the circuit would be reduced in the 40×80 , if not eliminated. Furthermore, the 40×80 airspeed has been increased from 103 m/sec to 155 m/sec since the study described here was accomplished.

The multiple-sideline source-location method applied to this wind-tunnel study required substantial commitments of time and effort such as (1) static, far field noise surveys of the engine outside the wind tunnel, (2) acoustic treatment of the test section walls, and (3) development of a complicated data-reduction scheme. Although a larger test section with a good acoustic environment should reduce these requirements, evaluation of alternative methods is recommended. Hoglund, for example, proposed a simple method using cross correlation for source location (ref. 29). Nonetheless, the source-location method described here does work, within the limits discussed above. The method used in this study resulted in a better understanding of the complicated roles played by jet and internal-engine noise sources during aircraft flight.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, CA 94035-1000, June 26, 1990

REFERENCES

1. Elias, G.: Essais du Motor J85 a S1 MA. Analyse de la Turbulence du Jet par Radiometrie Infrarouge et Correlations Infrarouge-Son. (Test of the J85 Engine at S1 MA. Analysis of Jet Turbulence by Infrared Radiometer and Infrared/Sound Correlations). In French. ONERA Rapport Technique No. 4/3353 PN, 25 June 1980.
2. Hoglund, L. E.: Static Source Locations for Four Nozzles Mounted on a J85 Engine. NASA CR-152401, 1979.
3. Elias, G.; and Schmidt, C.: Aerodynamic and Acoustic Comparisons of Scaled-Down and Actual Engine Jets. AIAA Paper 79-0574. AIAA 5th Aeroacoustics Conference, Mar. 1979.
4. Burley, R. R.; Karabinus, F. J.; and Freedman, R. J.: Flight Investigation of Acoustic and Thrust Characteristics of Several Exhaust Nozzles Installed on Underwing Nacelles on an F106 Airplane. NASA TM X-2854, 1973.
5. Antl, R. J.; and Burley, R. R.: Steady-State Airflow and Afterburning Performance Characteristics of Four J85-GE-13 Turbojet Engines. NASA TM X-1742, 1969.
6. Clapper, W. S.; and Stringas, E. J.: High Velocity Jet Noise Source Location and Reduction. Task 4-Development/Evaluation of Techniques for "Inflight" Investigation. FAA-RD-76-79, IV, Feb. 1977.
7. Hoch, R. G.: Some Thoughts on the Effects of Flight on Jet Noise as Observed in Actual Flight and in Wind Tunnels. J. Sound Vib., vol. 72, no. 3, 1980, pp. 379-389.
8. Drevet, P.; Duponchel, J. P.; and Jacques, J.: R. The Effect of Flight on Jet Noise as Observed on the Bertin Aerotrain. J. Sound Vib., vol. 54, no. 2, 1977, pp. 173-201. Also AIAA Paper 76-557, July 1976.
9. Hoch, R. G. and Berthelot, M.: Use of the Bertin Aerotrain for the Investigation of Flight Effects on Aircraft Engine Exhaust Noise. AIAA Paper 76-534. AIAA 3rd Aeroacoustics Conference, July 1976.
10. Duponchel, J. P. and Thomas, P.: Impact of Flight Effects on Multitube Suppressor Design. AIAA Paper 80-1029. AIAA 6th Aeroacoustics Conference, June 1980.
11. Atencio, A. Jr.: The Effect of Forward Speed on J85 Engine Noise from Suppressor Nozzles as Measured in the NASA Ames 40-by 80-Foot Wind Tunnel. NASA TND-8426, 1977.
12. Atencio, A. Jr.: Wind Tunnel Measurements of Forward Speed Effects on Jet Noise from Suppressor Nozzles and Comparison with Flight Test Data. AIAA Paper 75-870. AIAA 8th Fluid and Plasma Dynamics Conference, June 1975.
13. Beulke, M. R.; Clapper, W. S.; McCann, E. O.; and Morozumi, H. M.: A Forward Speed Effects Study on Jet Noise from Several Suppressor Nozzles in the NASA/Ames 40- by 80-Foot Wind Tunnel. NASA CR-114741, 1974.
14. Stone, J. R.; Miles, J. H.; and Sargent, N. B.: Effects of Forward Velocity on Noise for a J85 Turbojet Engine with Multi-Tube Suppressor from Wind Tunnel and Flight Tests. NASA TM X-73542, 1976.

15. Ahuja, K. K.; Tester, B. J.; and Tanna, H. K.: Calculation of Far-Field Jet Noise Spectra from Near-Field Measurements Using True Source Location. AIAA Paper 78-1153. AIAA 11th Fluid and Plasma Dynamics Conference, July 1978.
16. Jaeck, C. L.: Static and Wind Tunnel Near-Field/Far-Field Jet Noise Measurements from Model Scale Single-Flow Baseline and Suppressor Nozzles. Volume 1: Noise Source Locations and Extrapolation of Static Free-Field Jet Noise Data. NASA CR-137913, 1976.
17. Jaeck, C. L.: Static and Wind Tunnel Near-Field/Far-Field Jet Noise Measurements from Model Scale Single-Flow Baseline and Suppressor Nozzles. Volume 2: Forward Speed Effects. NASA CR-137914, 1976.
18. Jaeck, C. L.: Static and Wind Tunnel Near-Field/Far-Field Jet Noise Measurements from Model Scale Single-Flow Baseline and Suppressor Nozzles. Summary Report --- conducted in the Boeing Large Anechoic Test Chamber and the NASA Ames 40- by 80-Foot Wind Tunnel. NASA CR-2841, 1977.
19. Strout, F. G.; and Atencio, A., Jr.: Flight Effects on JT8D Engine Jet Noise as Measured in the NASA Ames 40- by 80-Foot Wind Tunnel. AIAA Paper 76-556. AIAA 3rd Aeroacoustics Conference, July 1976.
20. Strout, F. G.; and Atencio, A., Jr.: Flight Effects on Noise Generated by a JT8D Engine with Inverted Primary/Fan Flow. J Aircraft, vol. 17, no. 1, Jan. 1980, pp. 13-19. Also AIAA Paper 79-0614, Mar. 1979.
21. Strout, F. G. : Flight Effects on Noise Generated by the JT8D-17 Engine in a Quiet Nacelle and a Conventional Nacelle as Measured in the NASA Ames 40- by 80-Foot Wind Tunnel. NASA CR-2576, 1976.
22. Christophe, J.: The ONERA Wind Tunnels at Modane Centre and at Le Fauga Centre and Their Utilization in Subsonic Range. ONERA TP 1983-28, presented at 19th Subsonic Aerodynamic Testing Association Meeting, Apr. 1983.
23. Soderman, P. T.; and Noble, S. C.: Directional Microphone Array for Acoustic Studies of Wind Tunnel Models. J. Aircraft, vol. 12, no. 3, Mar. 1975, pp. 168-173.
24. Stone, J. R.; Groesbeck, D. E.; and Zola, C. L.: Conventional Profile Coaxial Jet Noise Prediction. AIAA Journal, vol. 21, no. 3, Mar. 1983, pp. 336-342.
25. Couratin, J.: Mesures Acoustiques en Soufflerie a Veine Guidee: Convention NASA-Ames/ONERA- 1ere Partie- Mise en Oeuvre des Logiciels d'Exploitation. (Acoustic Measurements in a Closed Test Section Wind Tunnel: NASA Ames/ONERA Cooperative Program -1st Part- Operation of the Data Reduction Programs). In French. ONERA Rapport Technique No. 23/3408 PN, 16 Dec. 1981.
26. Stone, J. R.: Prediction of In-Flight Exhaust Noise for Turbojet and Turbofan Engines. Noise Control Eng., vol. 10, no. 1, pp. 40-46, Jan.-Feb. 1978.
27. Olson, L. E; Zell, P. T.; Soderman, P. T.; Falarski, M. D.; Corsiglia, V. R.; and Edenborough, H. K.: Aerodynamic Flow Quality and Acoustic Characteristics of the 40- by 80-Foot Test Section Circuit of the National Full-Scale Aerodynamic Complex. SAE TP 872328, Intl.. Powered Lift Conf. & Expos., Santa Clara, CA, Dec. 1987.
28. Soderman, P. T.: Sources and Levels of Background Noise in the NASA Ames 40- by 80-Foot Wind Tunnel - A Status Report. NASA TM-100077, 1988.
29. Hoglund, L. E.: The Use of Cross-Correlations in Place of Multiple Sidelines for Source Location in Jet Noise Measurement. AIAA Paper 79-0618. AIAA 5th Aeroacoustics Conference, Mar. 1979.

Table 1. Jet exhaust conditions at the exit of the primary nozzle.^a

Wind speed V_a , m/sec	Jet exhaust speed V_j , m/sec	Jet total temperature T_o , °C	Engine rotation speed, rev/min	Jet pressure ^b ratio, P_j/P_a	Ideal ^c jet thrust, N
0	545	680	16,137	1.82	11,336
49	522	622	16,087	1.78	10,727
72	555	648	16,137	1.91	11,941
88	577	660	16,170	2.0	12,661
113	597	670	16,170	2.10	13,205
130	607	675	16,170	2.14	13,208

^aJet conditions computed from fluid mechanic measurements of pressure and temperature. Parameters correspond to conditions at altitude of S1 Wind Tunnel (1100 m).

^bRatio of jet total pressure to wind tunnel static pressure.

^cMomentum or ideal thrust equals net thrust plus ram drag. The maximum rated net thrust is 12,100 N.

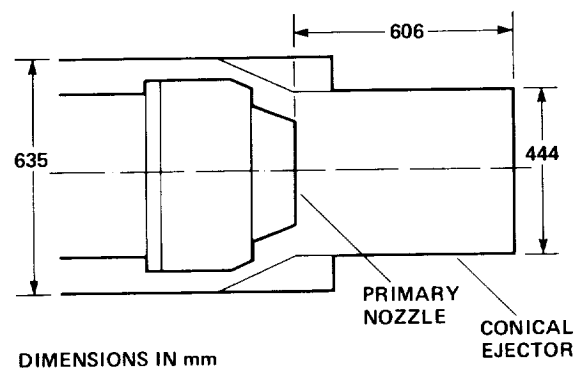
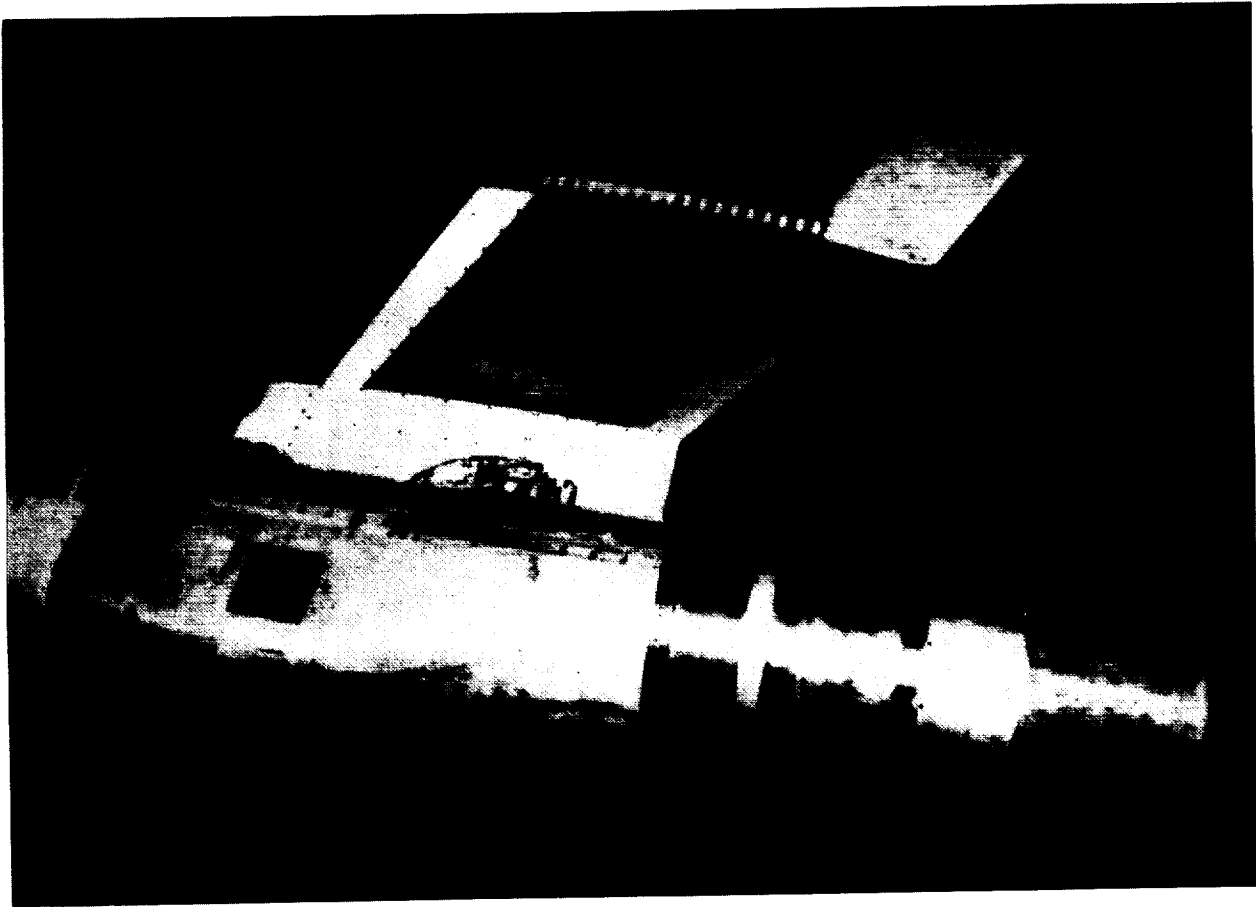
Table 2. Correction factors added to S1 data to account for amplification caused by propagation around the circuit.^a

θ , deg	Overall	ΔL_p , dB					
		125 Hz	250 Hz	500 Hz	1250 Hz	2500 Hz	5000 Hz
40	-3.0	-8.0	-7.5	-2.5	-1.8	-1.7	-0.9
50	-2.9	-6.0	-7.0	-2.0	-1.7	-1.7	-0.9
60	-2.3	-5.0	-5.0	-1.6	-1.7	-1.7	-0.9
70	-2.0	-4.5	-3.5	-1.5	-1.6	-1.7	-0.9
80	-1.7	-4.0	-3.0	-1.5	-1.6	-1.7	-0.9
90	-1.3	-3.0	-2.3	-1.5	-1.6	-1.7	-0.9
100	-1.3	-2.0	-1.8	-1.0	-1.5	-2.0	-0.9
110	-1.3	-1.0	-1.5	-0.7	-1.5	-2.0	-1.5
120	-1.1	0	-1.0	-0.3	-1.0	-2.0	-2.7
130	-0.5	0	0	0	-1.0	-2.0	-4.3
135	0	0	0	0	-1.0	-2.0	-4.2
140	0	0	0	0	-1.0	-2.5	-4.1
145	0	0	0	0	-1.0	-3.0	-4.0
150	0	0	0	0	-1.0	-3.0	-4.0
155	0	0	0	0	-1.0	-3.0	-4.0
160	0	0	0	0	-1.0	-3.0	-3.0

^aFactors are tabulated by direction and frequency. Interpolation was used for intermediate frequencies.

Table 3. Typical data on 2.0-m sideline with corrections and extrapolation to 122-m sideline; static case (induced wind = 16 m/sec).

No.	θ_s	X/D	$\theta_1; \psi_1$	$\theta_2; \psi_2$	Near field (2 m) L_p at θ_s	Absorption correction	Near field correction	Corrected far field (122 m) L_p at ψ_s
1	40.00	0.98	45.66	40.08	115.63	0.12	1.70	81.50
2	42.40	1.08	49.42	42.50	116.35	0.12	1.83	82.35
3	44.80	1.67	57.38	44.97	117.48	0.11	1.93	83.59
4	47.20	2.10	65.10	47.43	117.80	0.11	2.03	84.02
5	49.60	2.17	69.49	49.86	117.89	0.10	2.11	84.19
6	52.00	2.11	72.44	52.27	117.99	0.10	2.23	84.41
7	54.40	2.01	74.71	54.68	118.11	0.10	2.34	84.65
8	56.80	1.89	76.62	57.07	118.24	0.09	2.45	84.89
9	59.20	1.77	78.35	59.47	118.38	0.09	2.56	85.14
10	61.60	1.66	80.06	61.87	118.54	0.09	2.66	85.40
11	64.00	1.57	81.87	64.26	118.73	0.09	2.75	85.68
12	66.40	1.49	83.82	66.66	118.95	0.09	2.83	85.99
13	68.80	1.45	86.07	69.06	119.22	0.08	2.87	86.30
14	71.20	1.44	88.71	71.47	119.55	0.08	2.91	86.67
15	73.60	1.49	91.92	73.88	119.95	0.08	2.93	87.09
16	76.00	1.64	96.30	76.32	120.43	0.08	2.95	87.60
17	78.40	1.93	102.33	78.78	120.96	0.08	2.97	88.14
18	80.80	2.18	107.63	81.24	121.32	0.08	2.98	88.52
19	83.20	2.27	110.78	83.66	121.55	0.08	2.99	88.76
20	85.60	2.27	112.85	86.07	121.73	0.08	3.00	88.95
21	88.00	2.22	114.34	88.46	121.89	0.08	3.00	89.10
22	90.40	2.14	115.55	90.84	122.03	0.08	3.00	89.25
23	92.80	2.05	116.60	93.22	122.17	0.08	3.00	89.39
24	95.20	1.97	117.65	95.60	122.33	0.08	3.00	89.54
25	97.60	1.87	118.62	97.98	122.49	0.08	2.99	89.69
26	100.00	1.79	119.70	100.36	122.68	0.08	2.98	89.88
27	102.40	1.72	120.90	102.74	122.93	0.08	2.97	90.11
28	104.80	1.66	122.19	105.12	123.22	0.08	2.95	90.38
29	107.20	1.62	123.64	107.50	123.60	0.08	2.92	90.73
30	109.60	1.59	125.22	109.89	124.05	0.08	2.90	91.16
31	112.00	1.58	126.94	112.28	124.61	0.08	2.86	91.68
32	114.40	1.59	128.78	114.67	125.28	0.09	2.80	92.30
33	116.80	1.62	130.73	117.07	126.07	0.09	2.72	92.99
34	119.20	1.66	132.77	119.46	126.95	0.09	2.62	93.78
35	121.60	1.73	134.88	121.86	127.92	0.09	2.52	94.64
36	124.00	1.81	137.03	124.26	128.94	0.10	2.42	95.56
37	126.40	1.92	139.23	126.66	129.98	0.10	2.30	96.48
38	128.80	2.05	141.43	129.06	130.97	0.10	2.19	97.35
39	131.20	2.20	143.64	131.46	131.85	0.10	2.09	98.13
40	133.60	2.37	145.83	133.86	132.57	0.11	2.00	98.75
41	136.00	2.56	147.99	136.25	133.05	0.11	1.90	99.13
42	138.40	2.78	150.09	138.65	133.26	0.12	1.79	99.22
43	140.80	3.02	152.12	141.05	133.17	0.12	1.65	98.98
44	143.20	3.25	154.01	143.44	132.78	0.13	1.48	98.43
45	145.60	3.45	155.75	145.83	132.15	0.14	1.31	97.62
46	148.00	3.57	157.26	148.21	131.39	0.15	1.12	96.65
47	150.40	3.61	158.63	150.58	130.52	0.16	0.91	95.56
48	152.80	3.57	159.90	152.95	129.56	0.17	0.70	94.38
49	155.20	3.45	161.12	155.32	128.52	0.19	0.54	93.16
50	157.60	3.25	162.34	157.70	127.36	0.21	0.37	91.81
f = 500 Hz			$St_0 = 0.404$ $St = 0.416$		Radial distance correction $20 \cdot \log(YY/AN) = 35.71$			

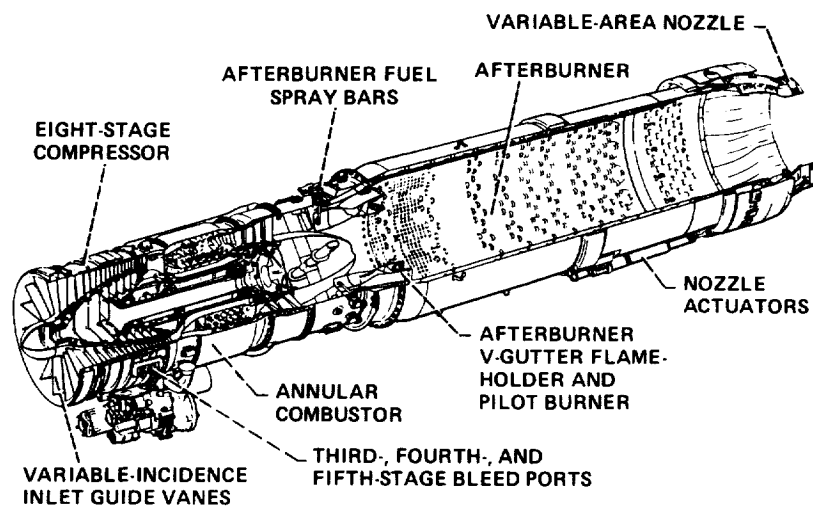


(a) Engine mounted in the S1 Wind Tunnel test section.

Figure 1. J-85 engine.

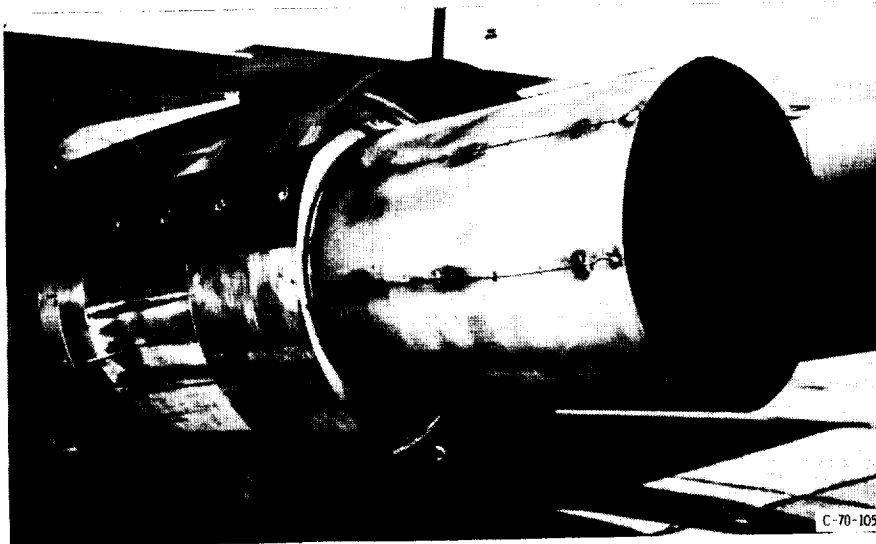
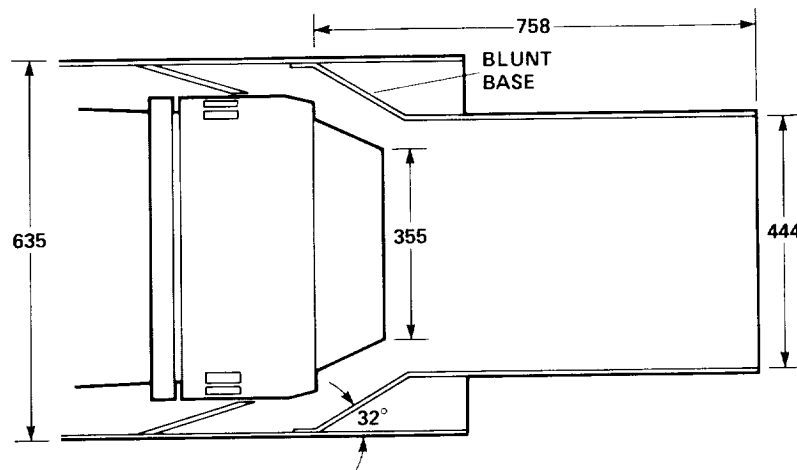
ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH



(b) Cutaway drawing of J-85-GE-13 afterburning turbojet engine (ref. 5).

Figure 1. Continued.



(c) Cylindrical ejector nozzle dimensions (above) and photo (below).

Figure 1. Concluded.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

ORIGINAL PAGE IS
OF POOR QUALITY

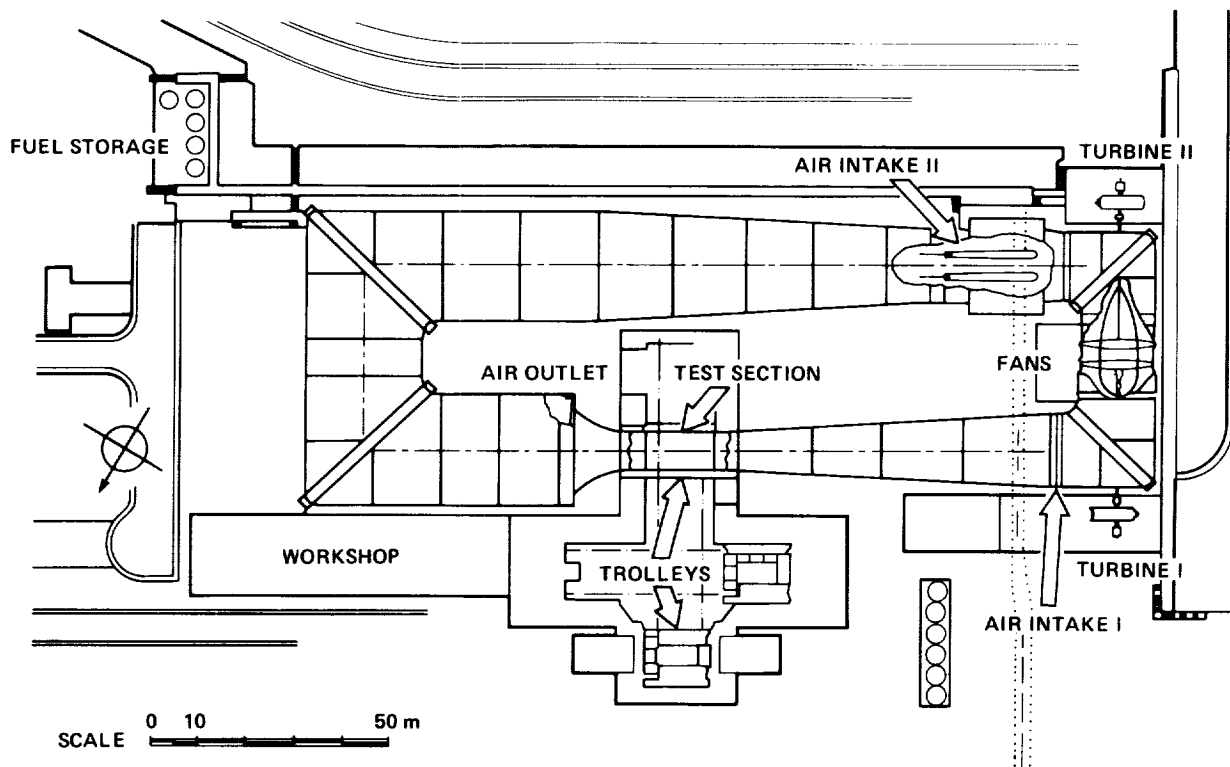


Figure 2. General layout of the S1 Wind Tunnel.

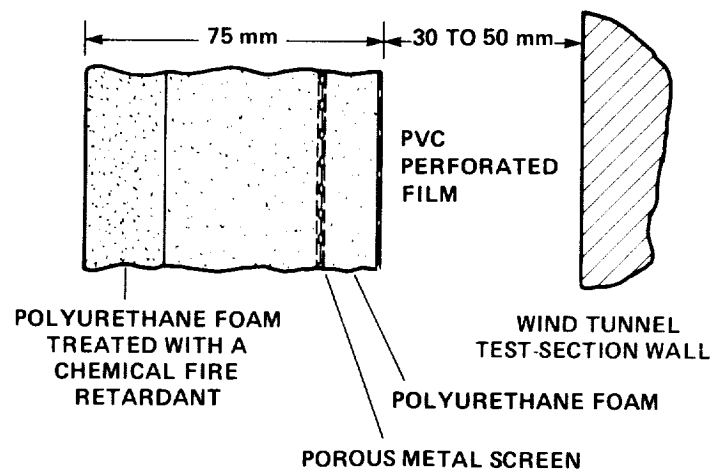


Figure 3. Cross section of the absorbent lining on the test section walls.

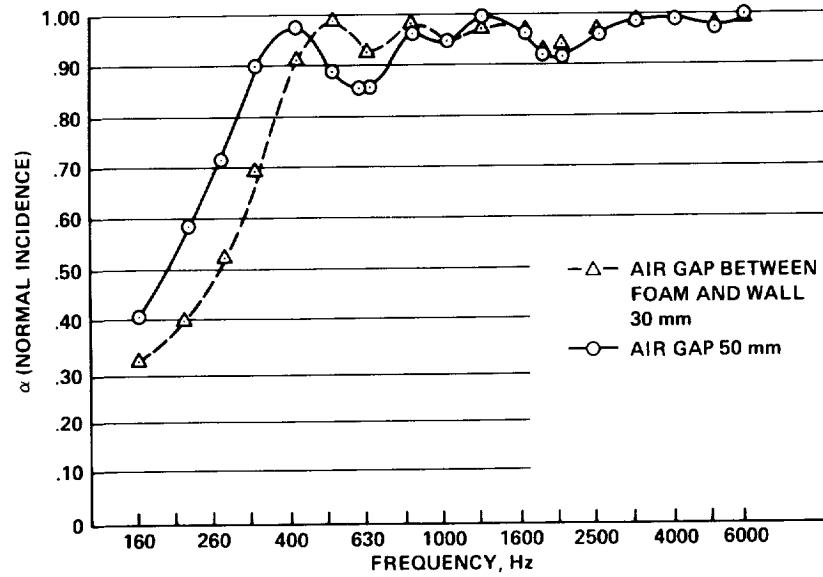
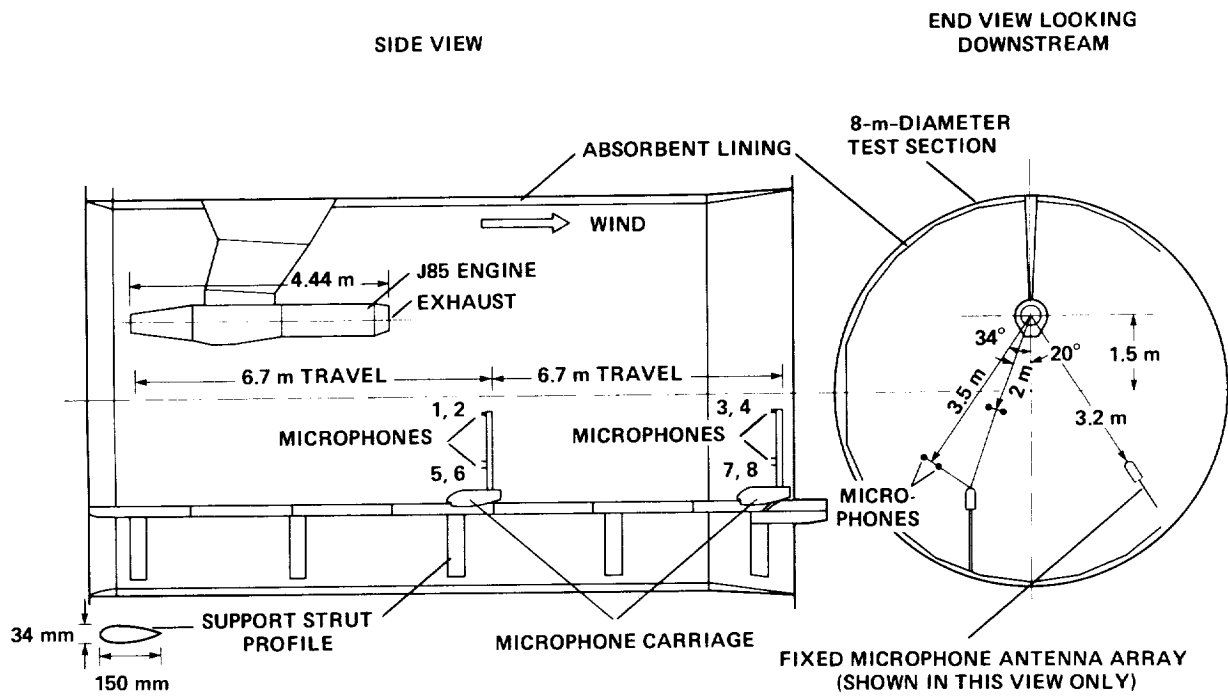


Figure 4. Acoustic absorption coefficients of the composite lining material (75 mm thick) plus air gap as measured in a standing wave tube.



(a) Assembly schematic.

Figure 5. Model and microphone systems in the S1 test section.

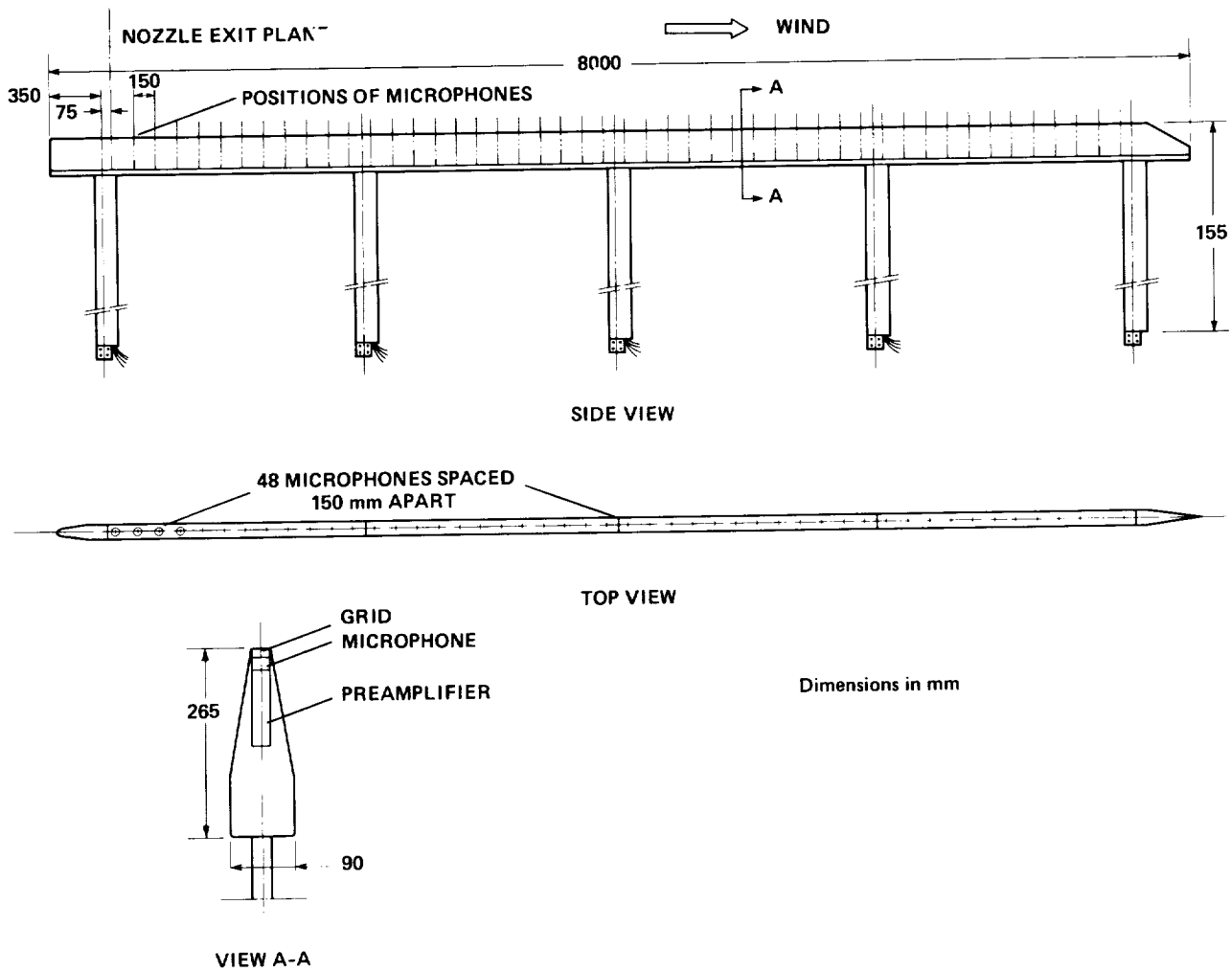


(b) Assembly photograph (traverse support strut fairings not installed yet).

Figure 5. Continued.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

ORIGINAL PAGE IS
OF POOR QUALITY



(c) Schematic of fixed microphone antenna array.

Figure 5. Concluded.

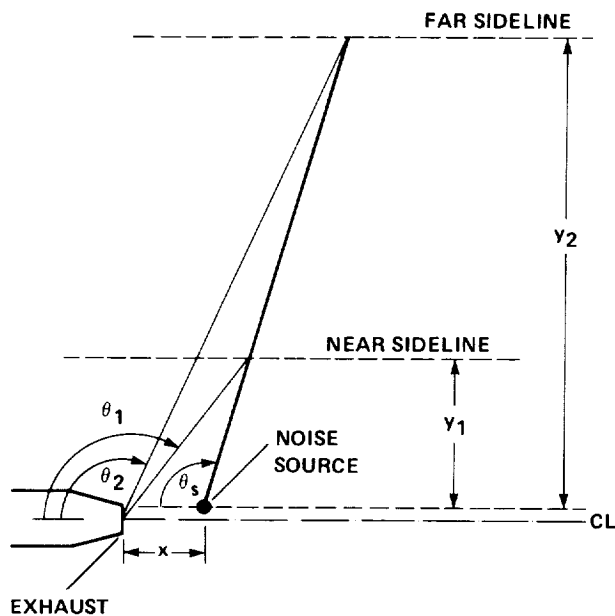


Figure 6. Source location geometry (static case).

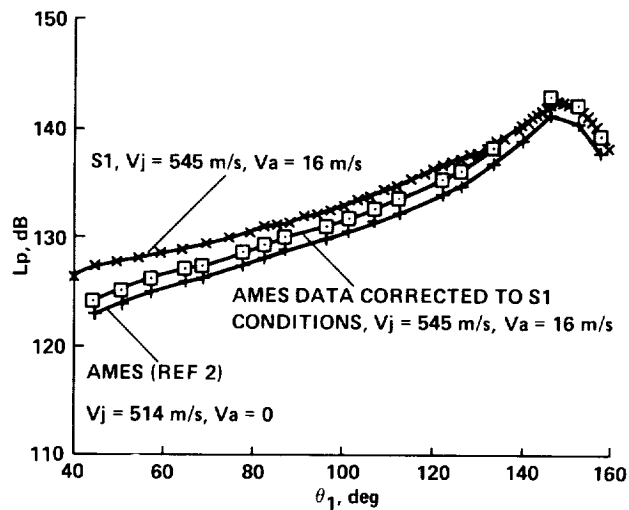


Figure 7. A comparison of J-85 noise on 2.0-m sideline measured in S1 Wind Tunnel and at NASA Ames outdoor test site (static case); overall sound levels.

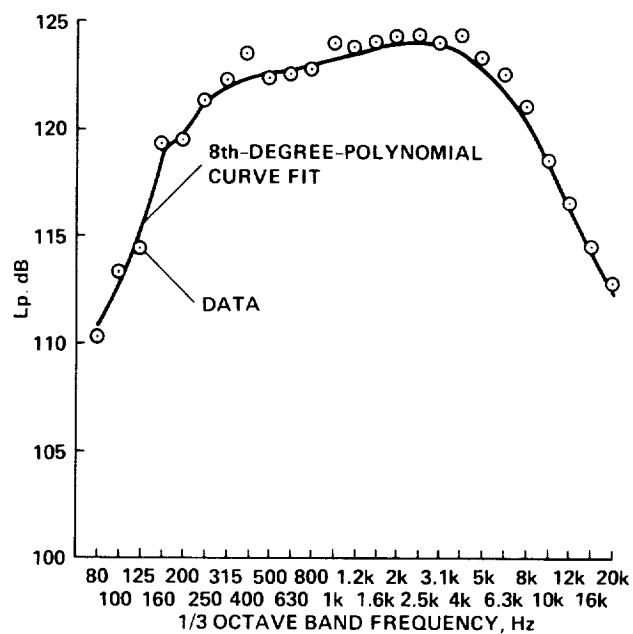
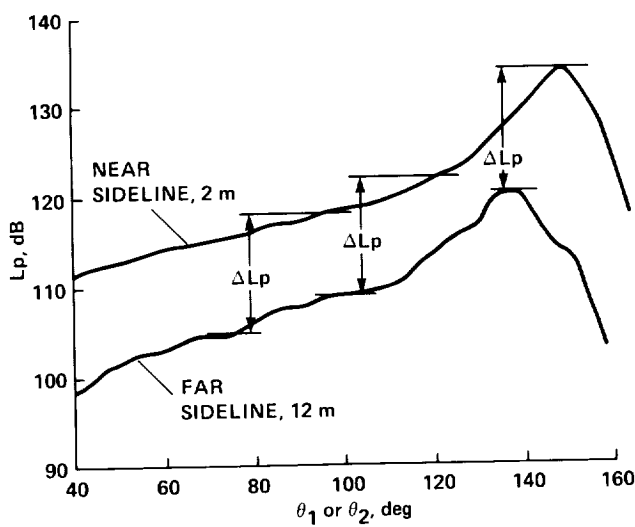
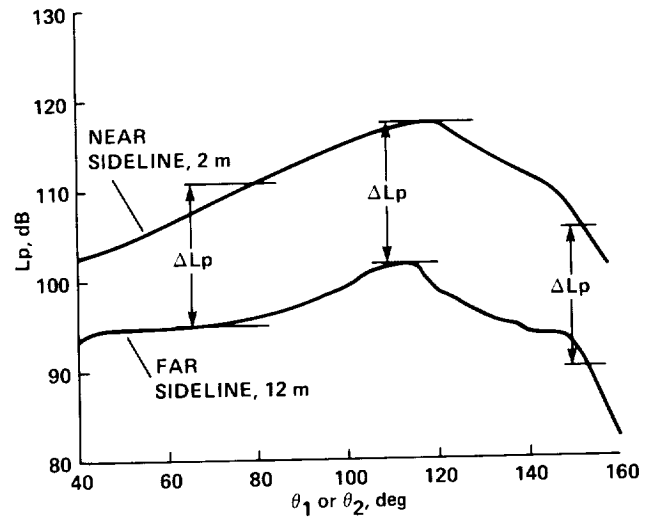


Figure 8. An example of 2.0-m sideline raw data and curve fit (static case); $\theta_1 = 119^\circ$.



(a) $f = 500\text{-Hz}$ third-octave band.



(b) 8-kHz third-octave band.

Figure 9. Typical angle pairs with the same noise difference as that for the peak levels (ref. 2).

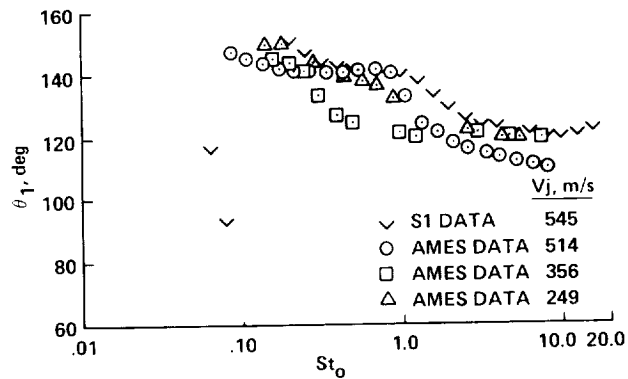
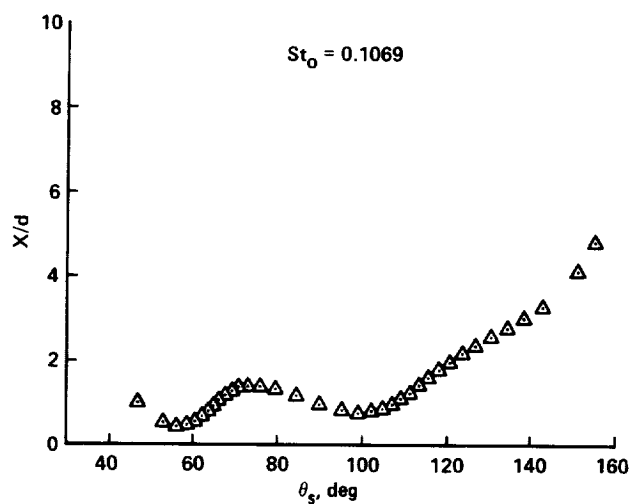
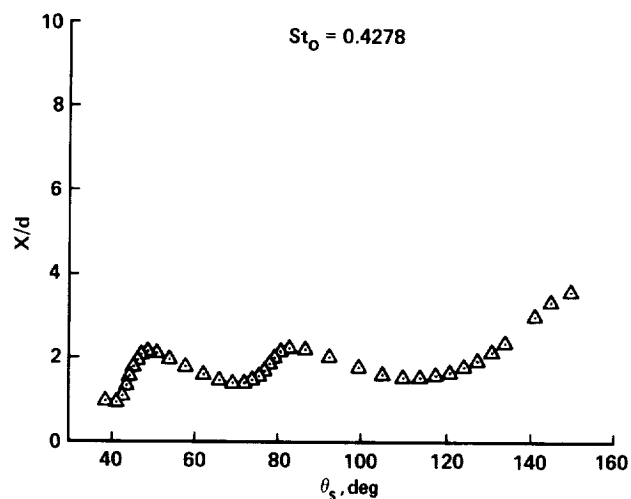


Figure 10. Angle of peak radiation versus Strouhal number St_o (static case) measured at S1 and at Ames (ref. 2).

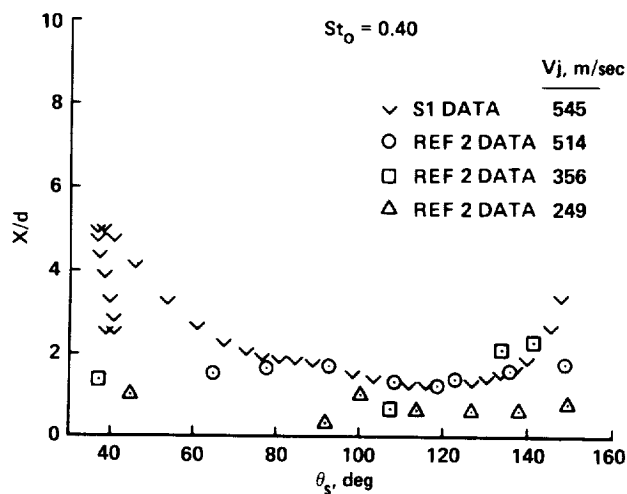


(a) $f = 125$ Hz.

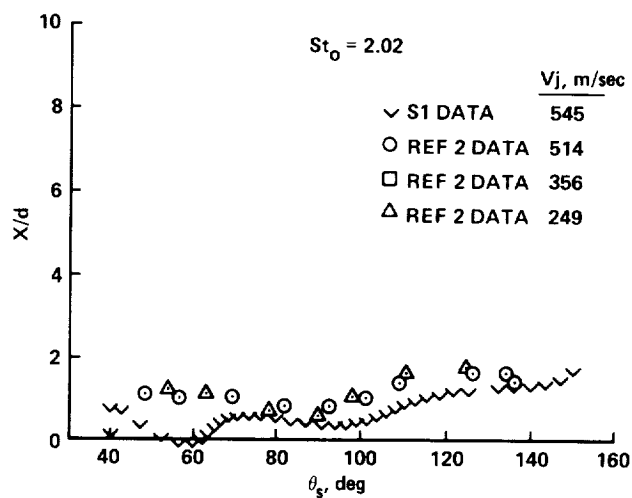


(b) $f = 500$ Hz.

Figure 11. Typical source locations versus radiation angle for the static case (ref. 2); $V_j = 514$ m/sec.



(a) $f = 500$ Hz.



(b) $f = 2500$ Hz.

Figure 12. A comparison of source locations measured statically (ref. 2) at Ames and in the S1 Wind Tunnel.

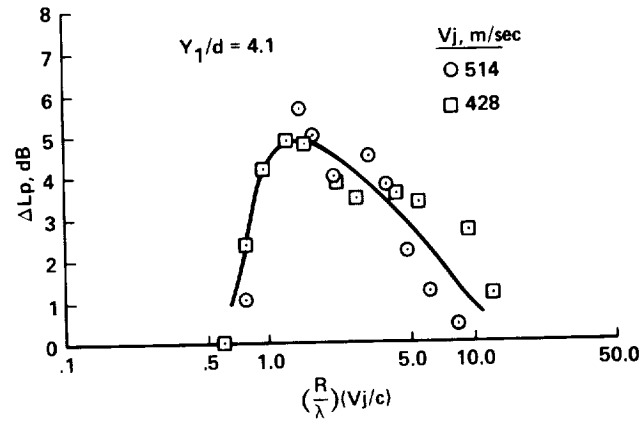


Figure 13. Near field corrections (ref. 2).

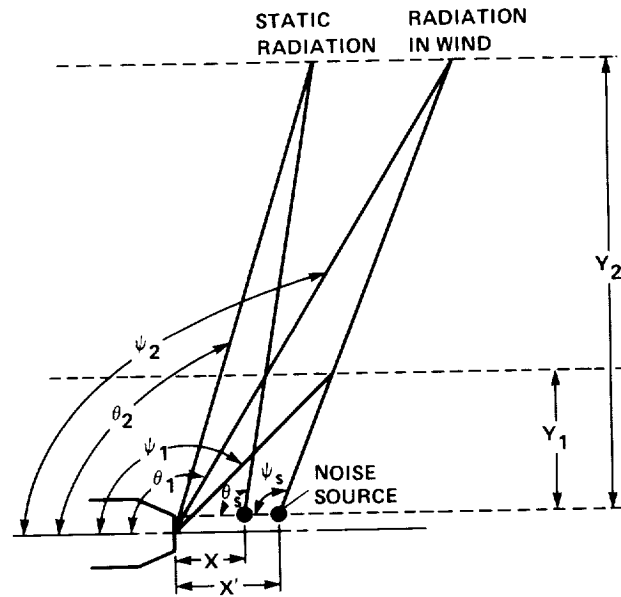
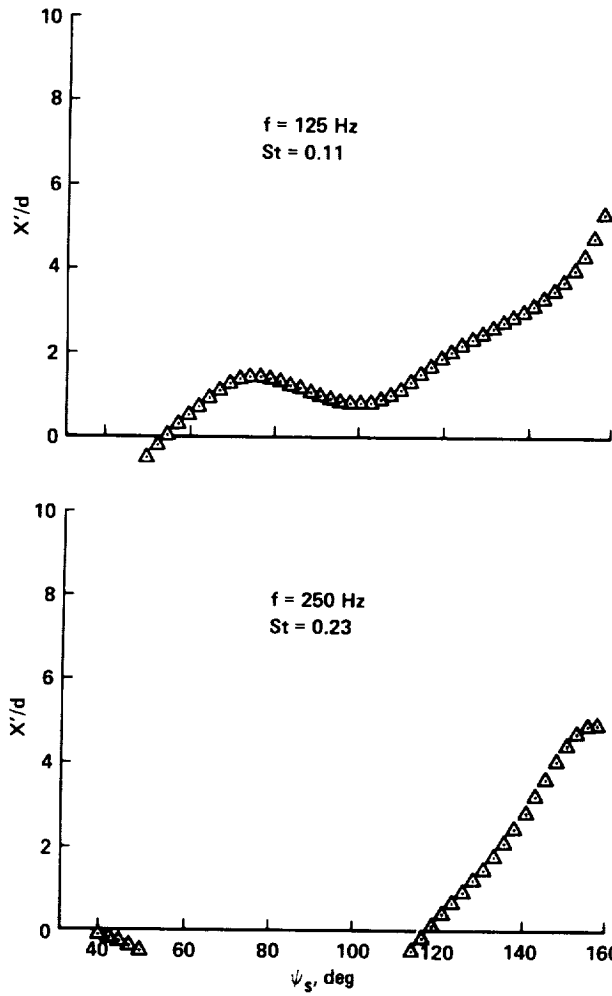
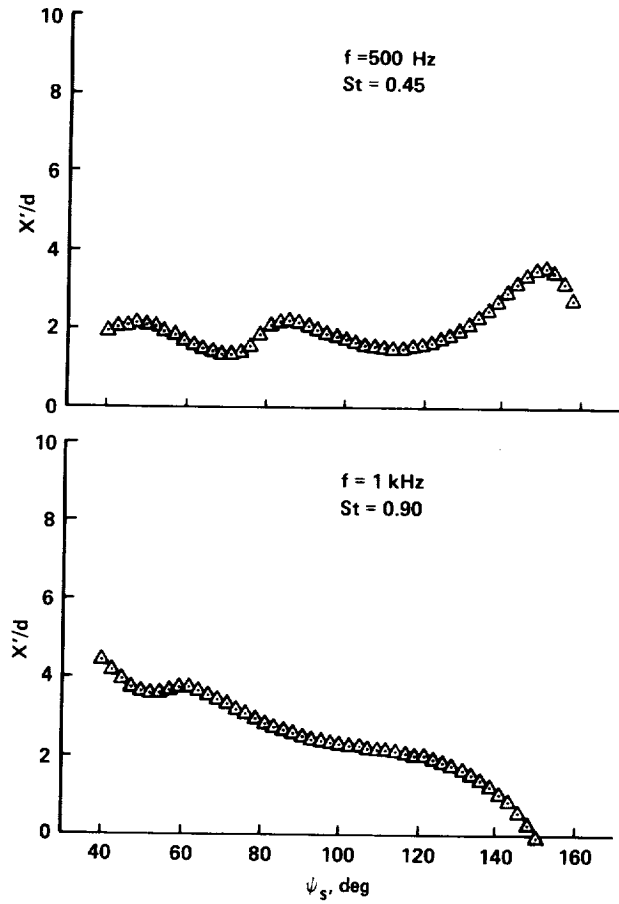


Figure 14. Nomenclature for source location and radiation with and without wind.

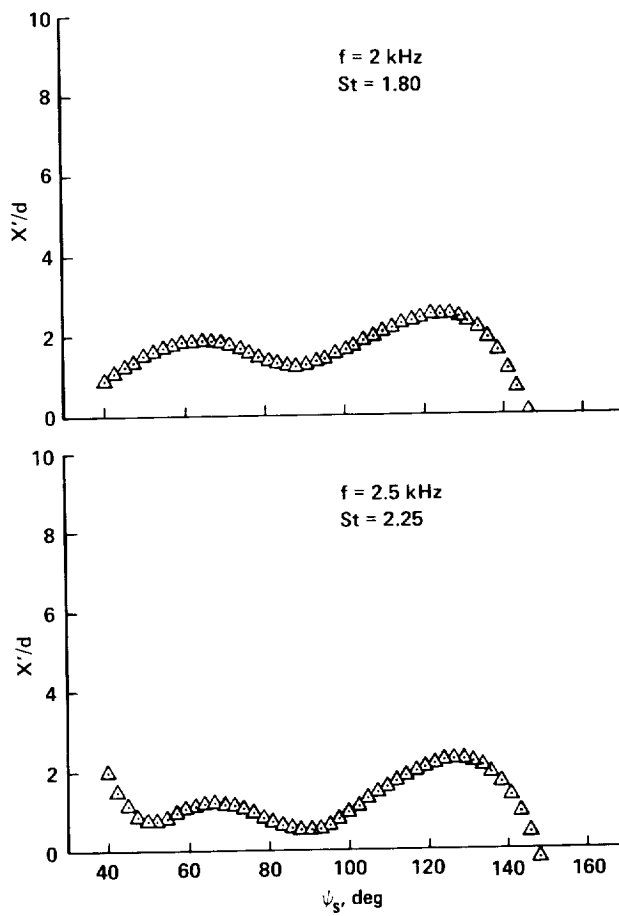


(a) $f = 125$ and 250 Hz.

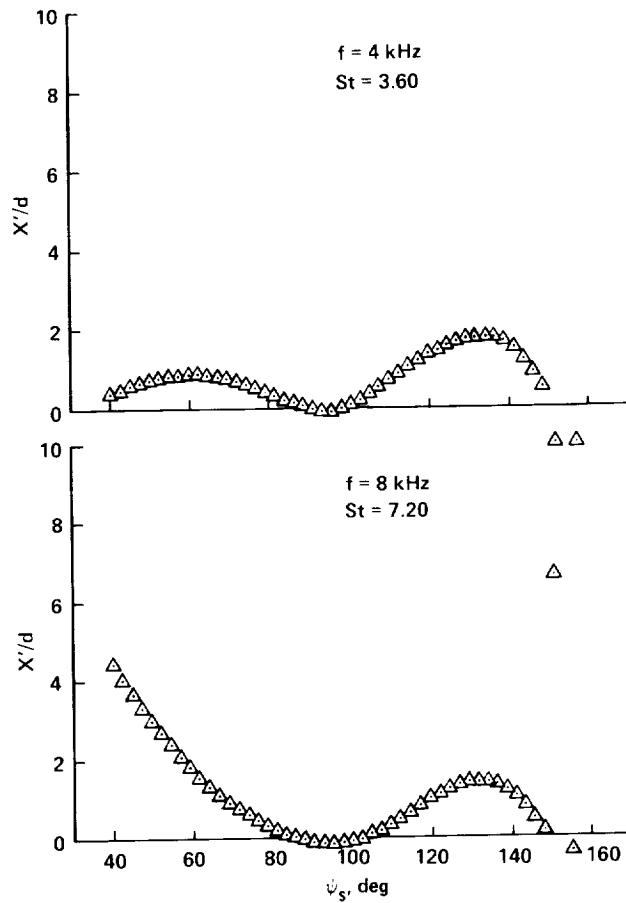


(b) $f = 500$ and 1 kHz.

Figure 15. Typical source locations versus radiation angle as transformed from static data using equations 1 and 2; $V_a = 88$ m/sec, $V_j = 577$ m/sec.

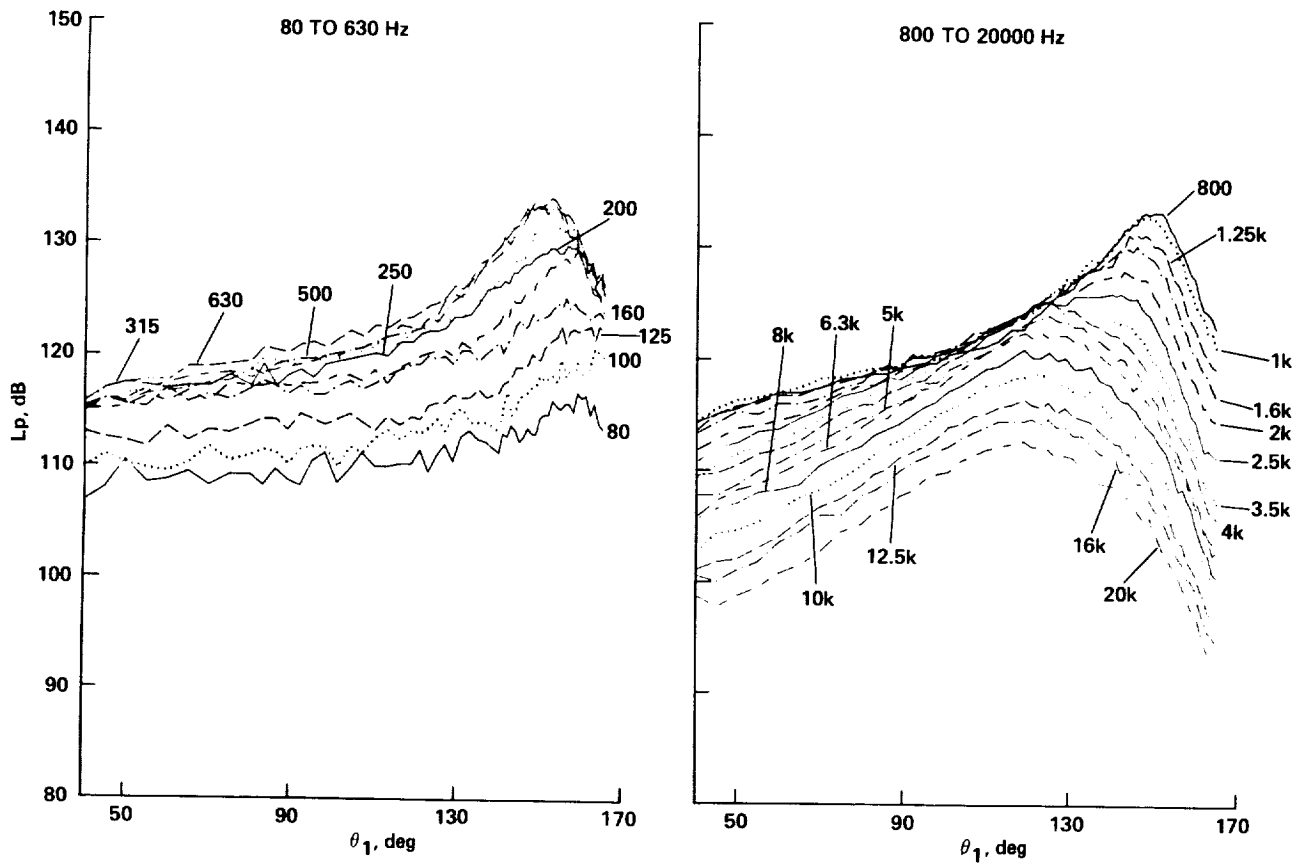


(c) $f = 2 \text{ kHz}$ and 2.5 kHz .



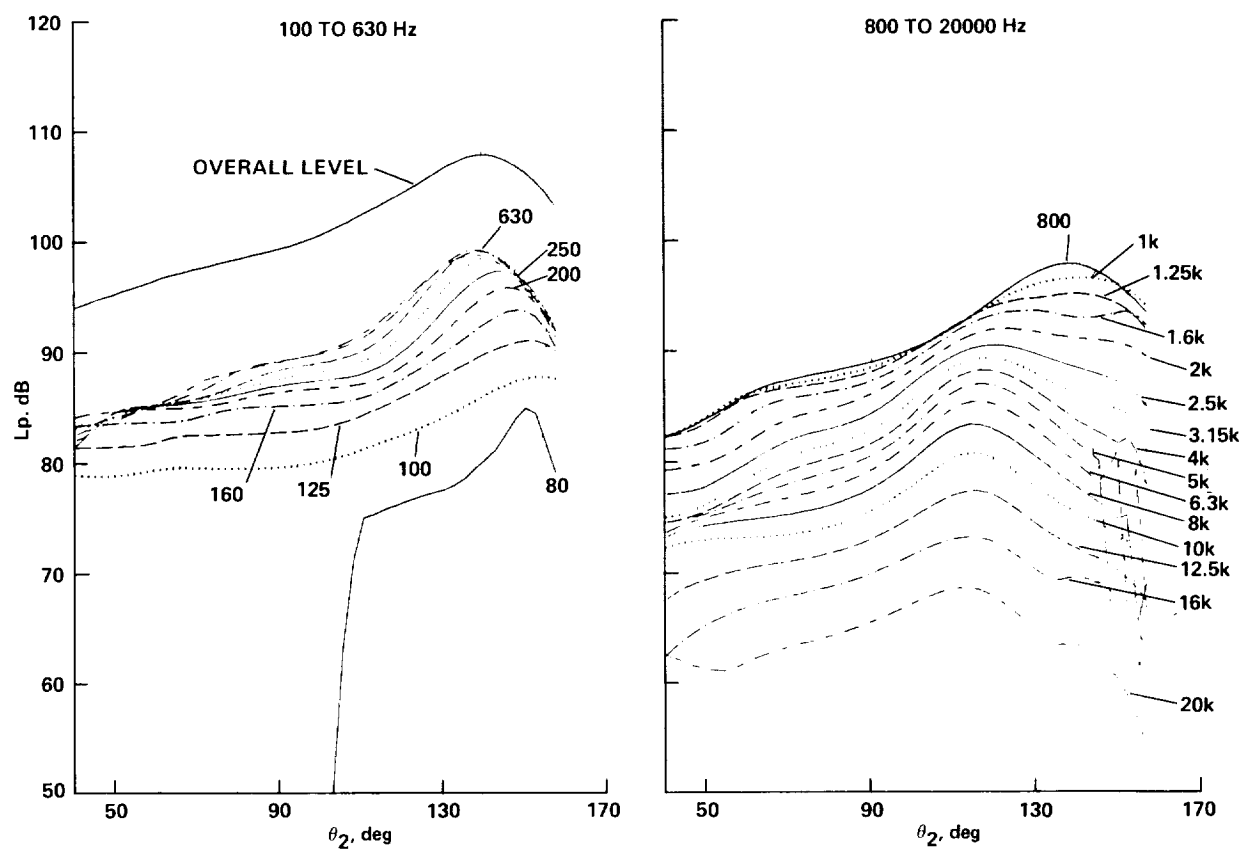
(d) $f = 4 \text{ kHz}$ and 8 kHz .

Figure 15. Concluded.



(a) 2.0 m data.

Figure 16. Static spectra at 2.0 m and extrapolated to 122-m sideline; $V_a = 16$ m/sec, $V_j = 545$ m/sec.



(b) Extrapolation to 122 m.

Figure 16. Concluded.

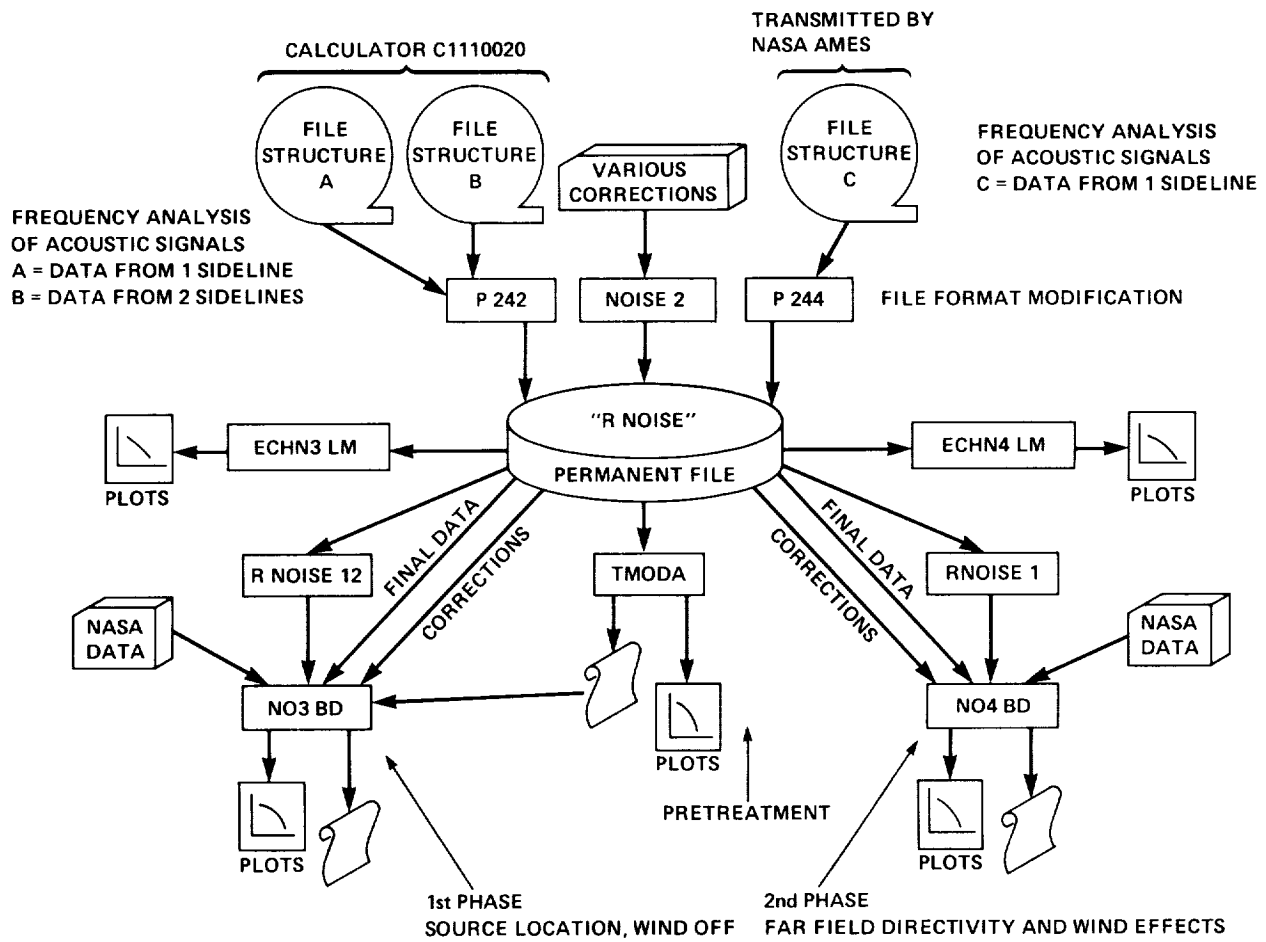
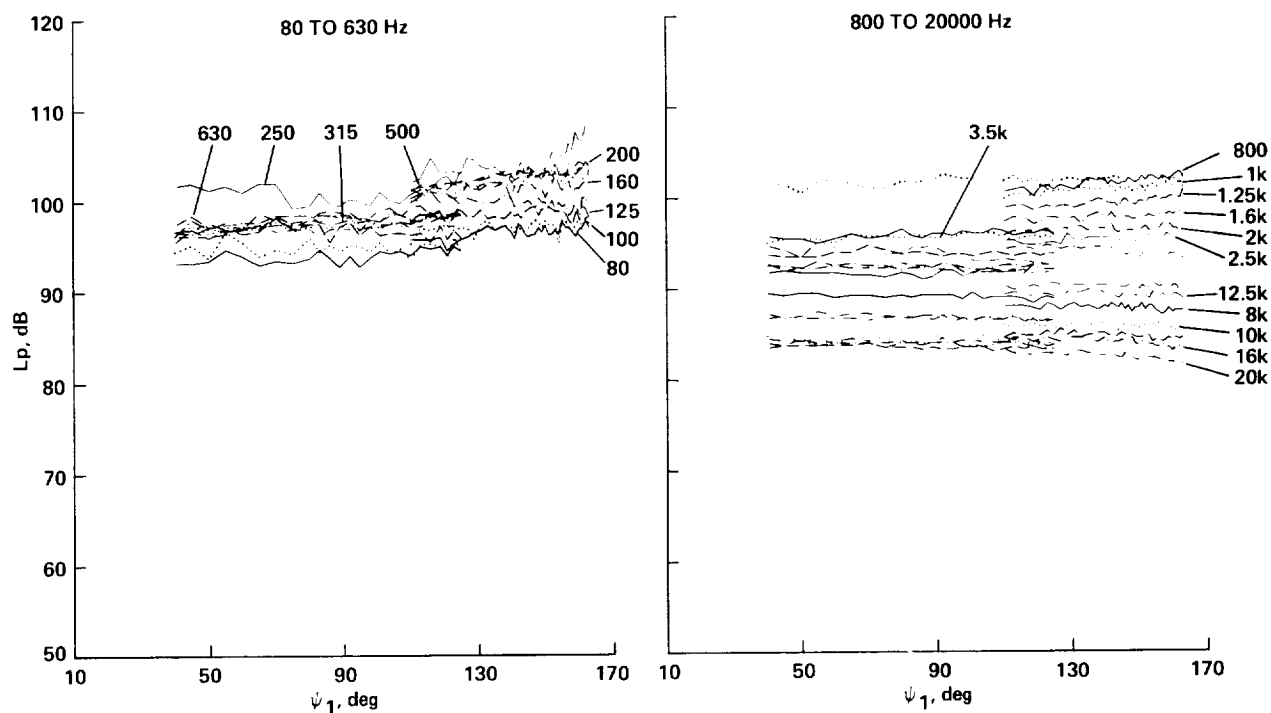
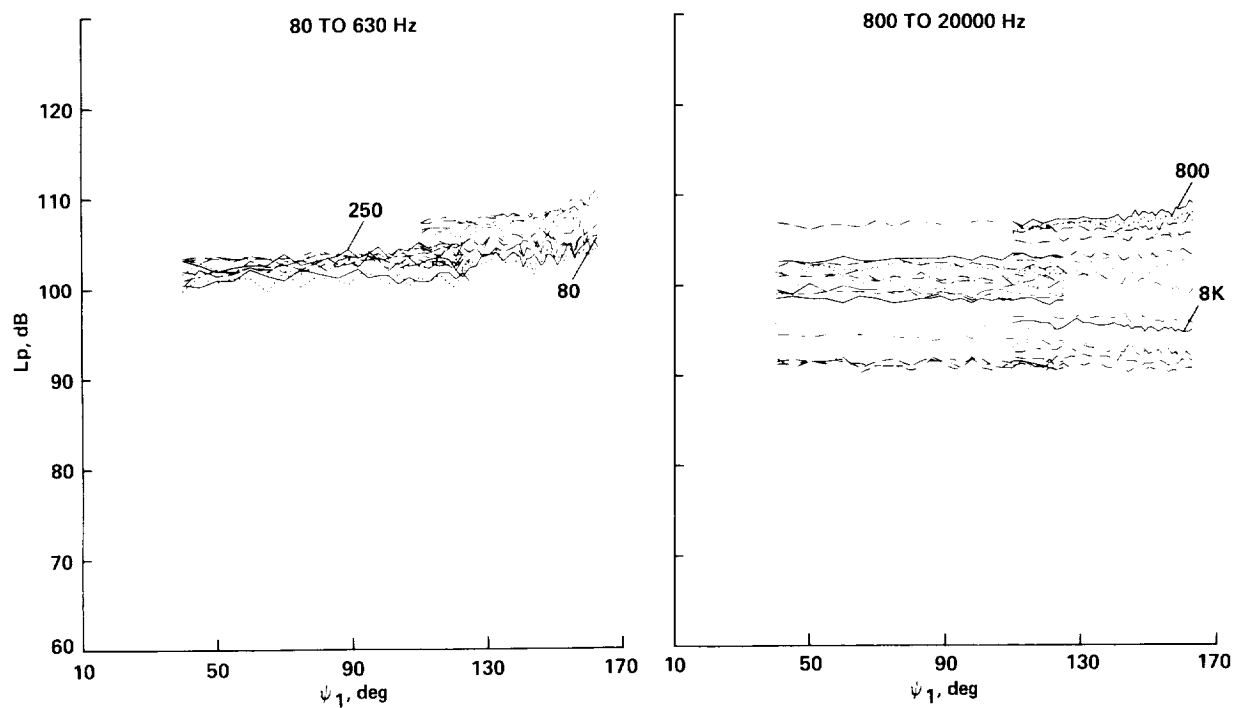


Figure 17. Schematic of data reduction procedures (ref. 23).

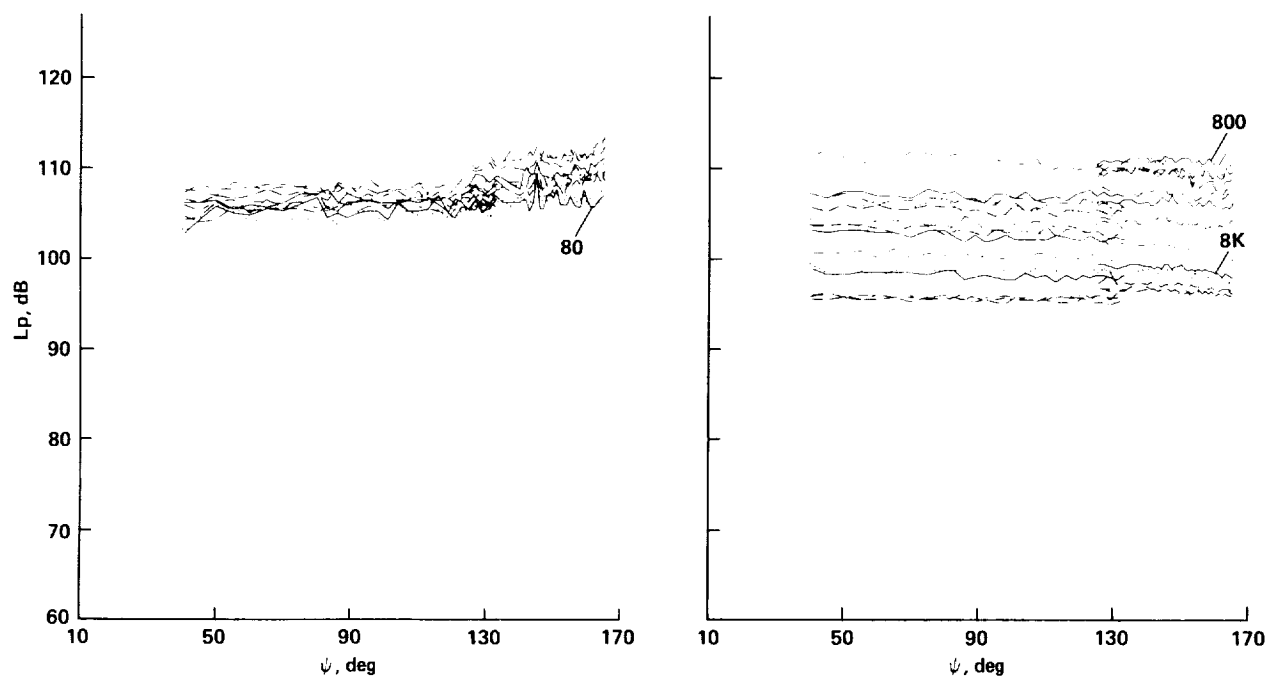


(a) $V_a = 87$ m/sec.



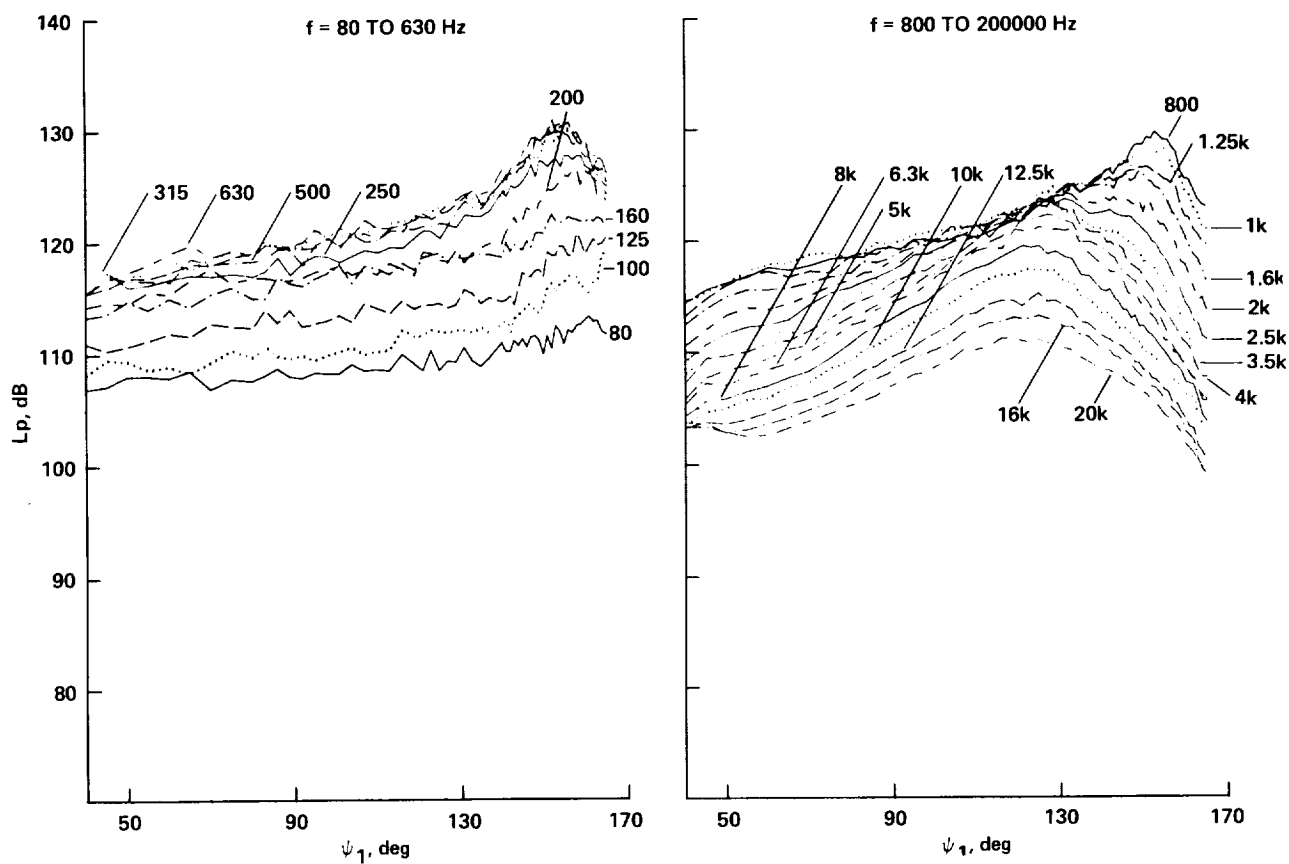
(b) $V_a = 113$ m/sec.

Figure 18. Background noise in S1 Wind Tunnel on 2.0-m sideline. J-85 in test section, but not operating.



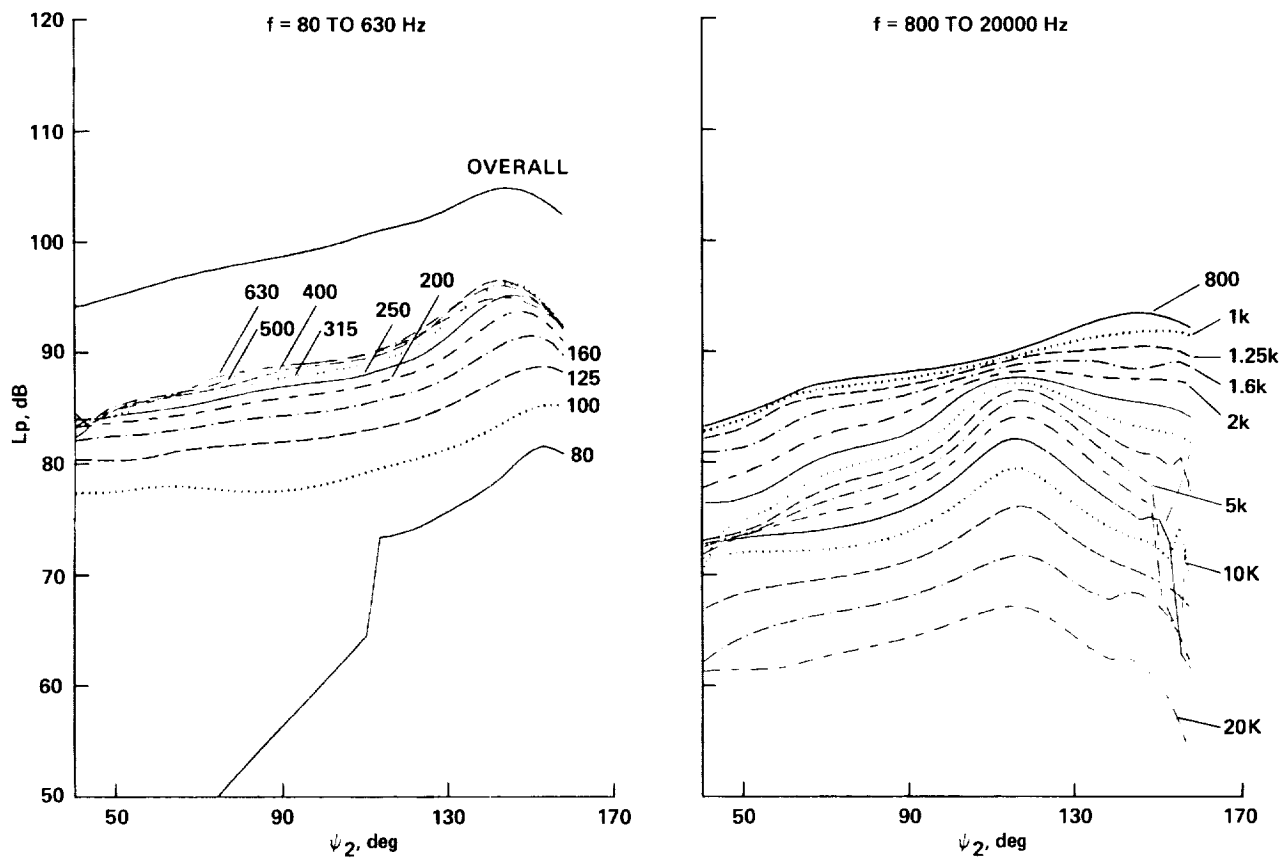
(c) $V_a = 130$ m/sec.

Figure 18. Concluded.



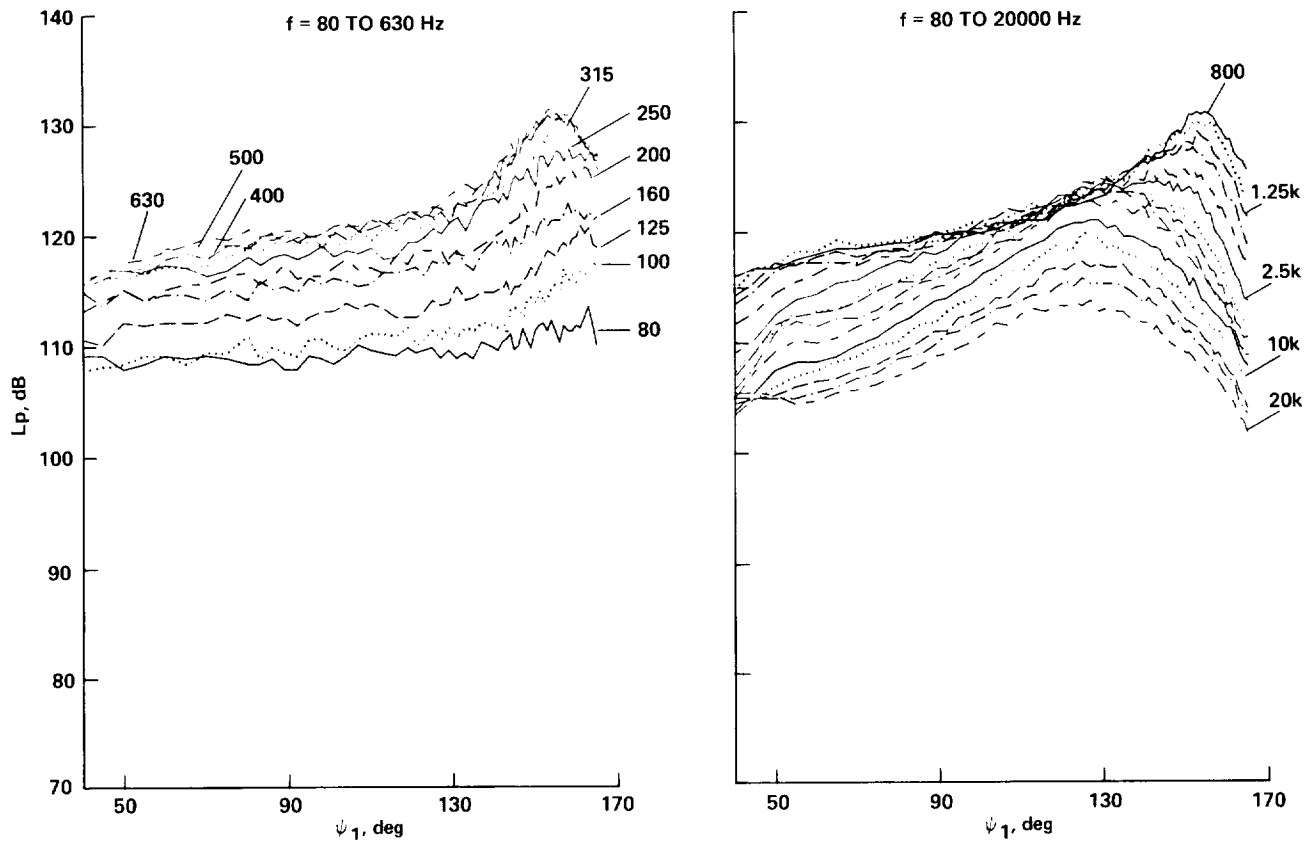
(a) $V_a = 49 \text{ m/sec}$, $V_j = 522 \text{ m/sec}$; 2.0-m sideline.

Figure 19. Acoustic spectra on 2.0-m sideline with extrapolations to 122-m sideline for several forward speeds.



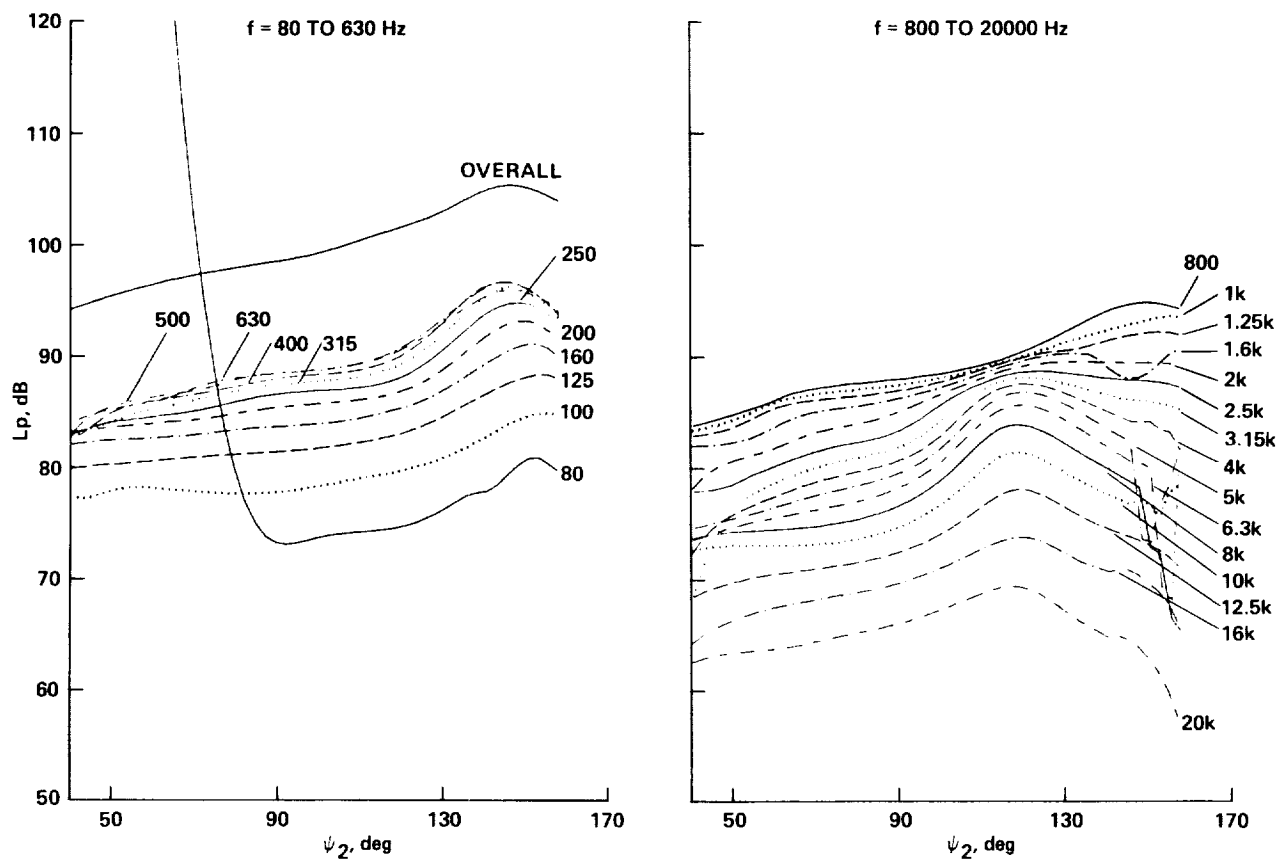
(b) $V_a = 49 \text{ m/sec}$, $V_j = 522 \text{ m/sec}$; 122-m sideline.

Figure 19. Continued.



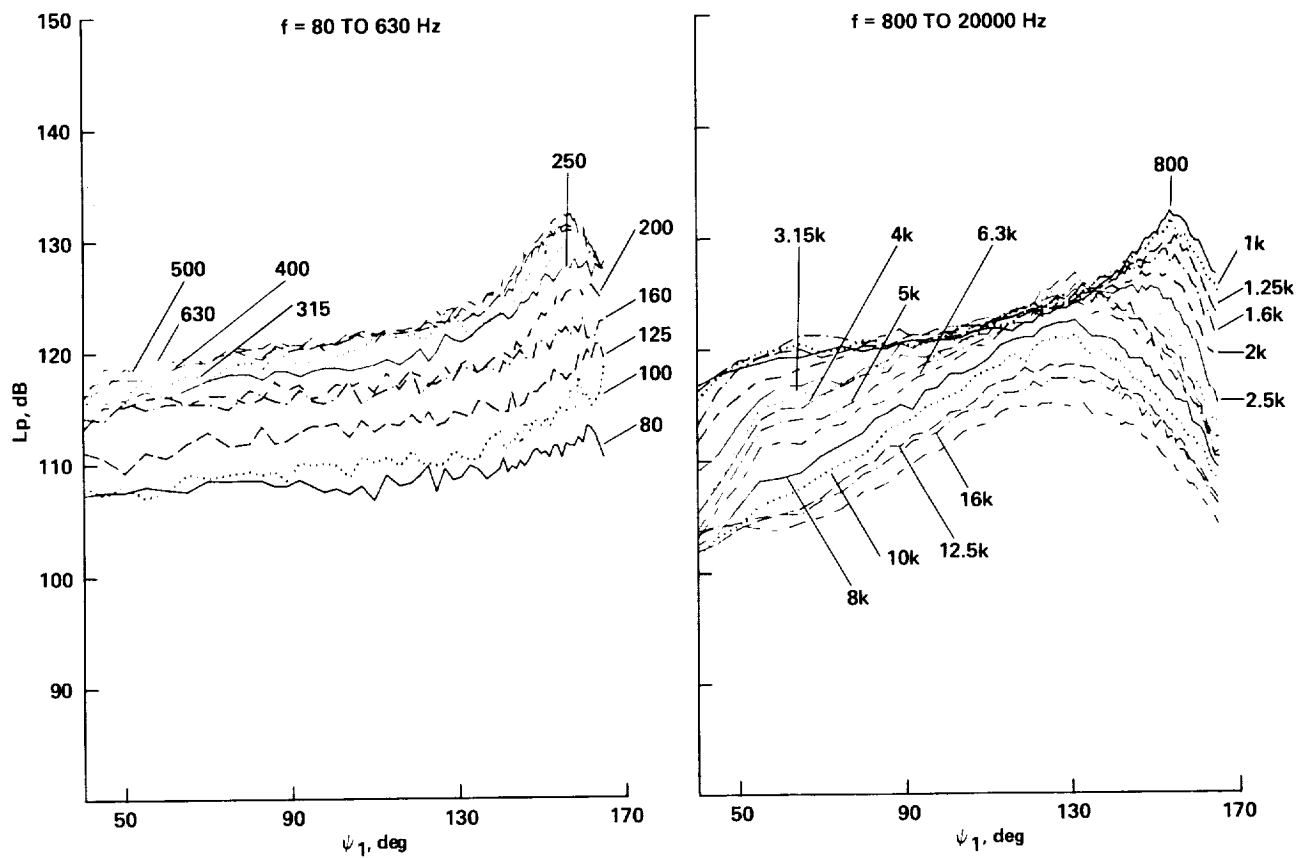
(c) $V_a = 72 \text{ m/sec}$, $V_j = 555 \text{ m/sec}$; 2.0-m sideline.

Figure 19. Continued.



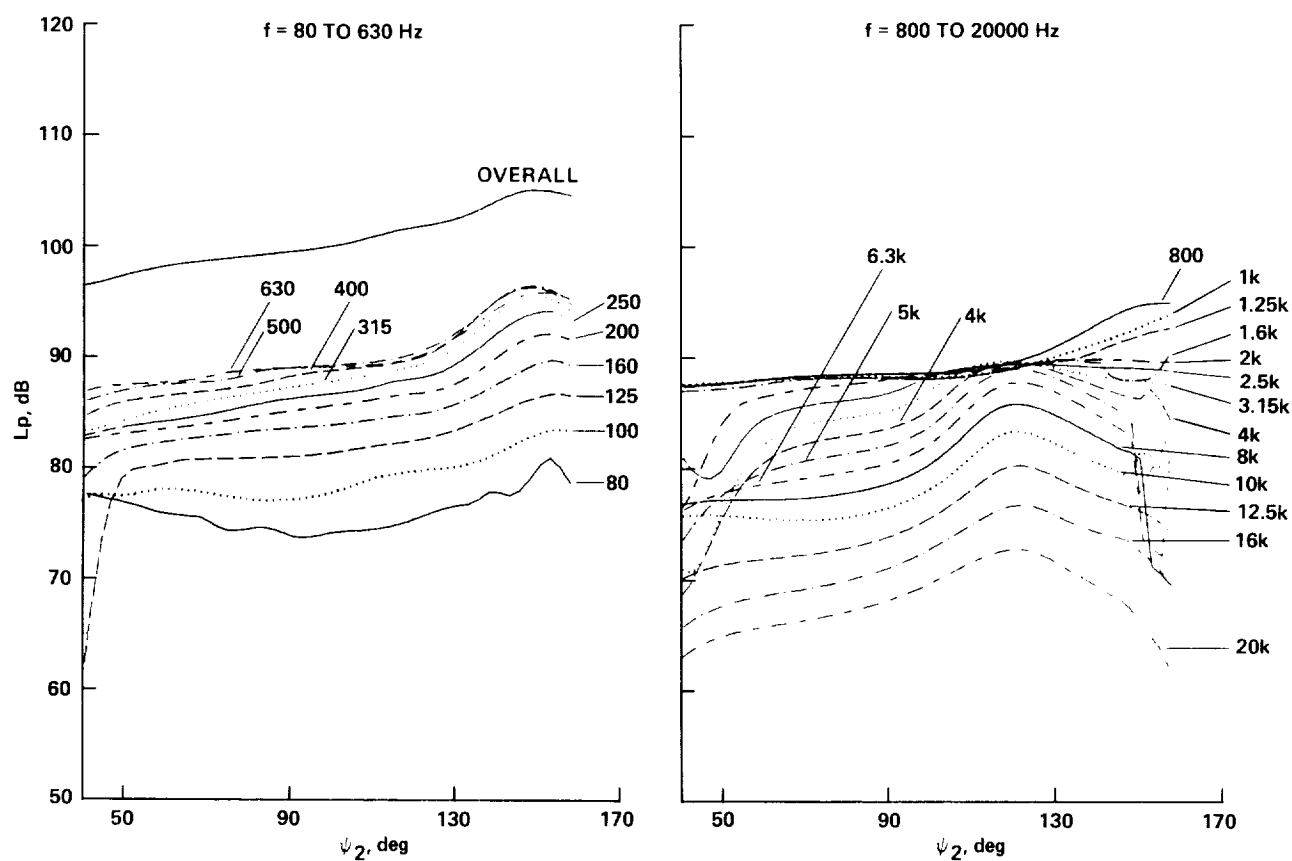
(d) $V_a = 72$ m/sec, $V_j = 555$ m/sec; 122-m sideline.

Figure 19. Continued.



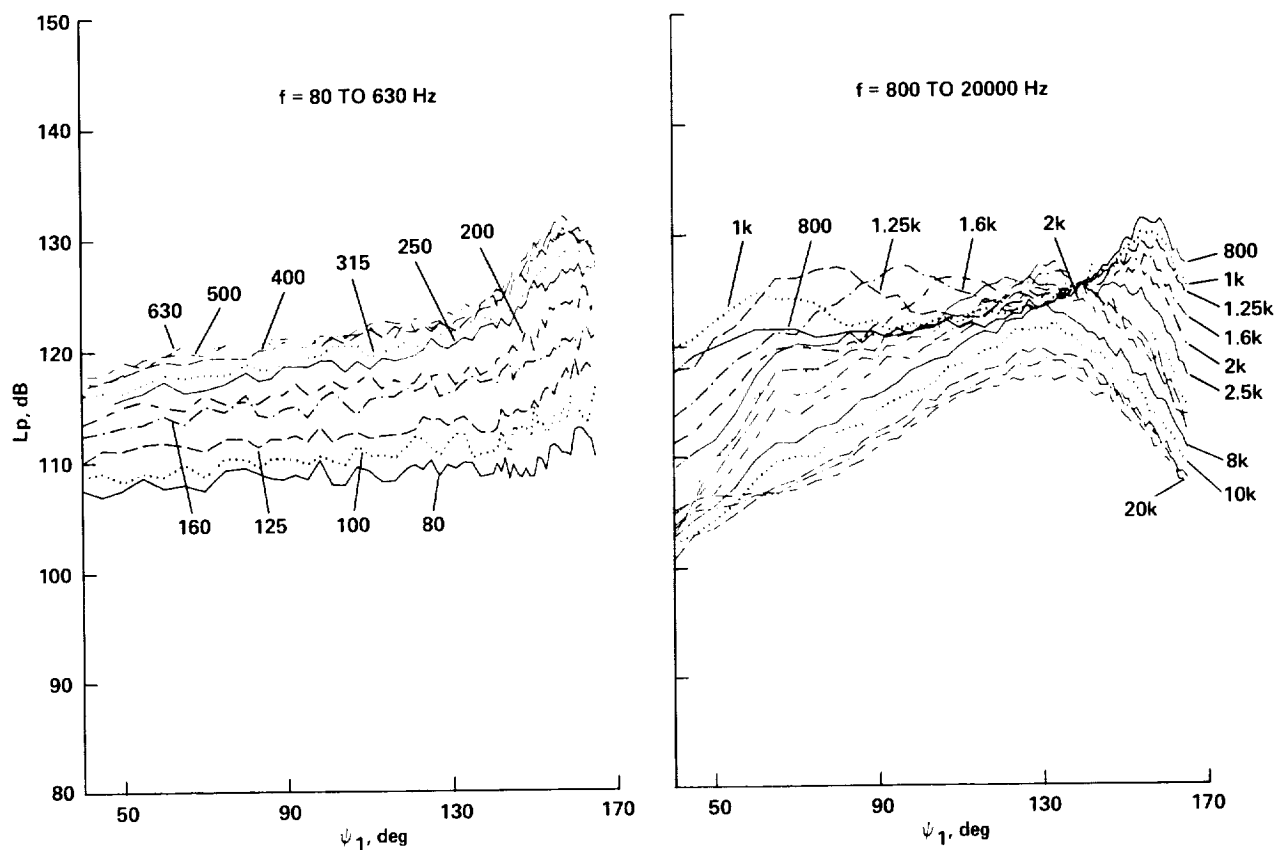
(e) $V_a = 88 \text{ m/sec}$, $V_j = 577 \text{ m/sec}$; 2.0-m sideline.

Figure 19. Continued.



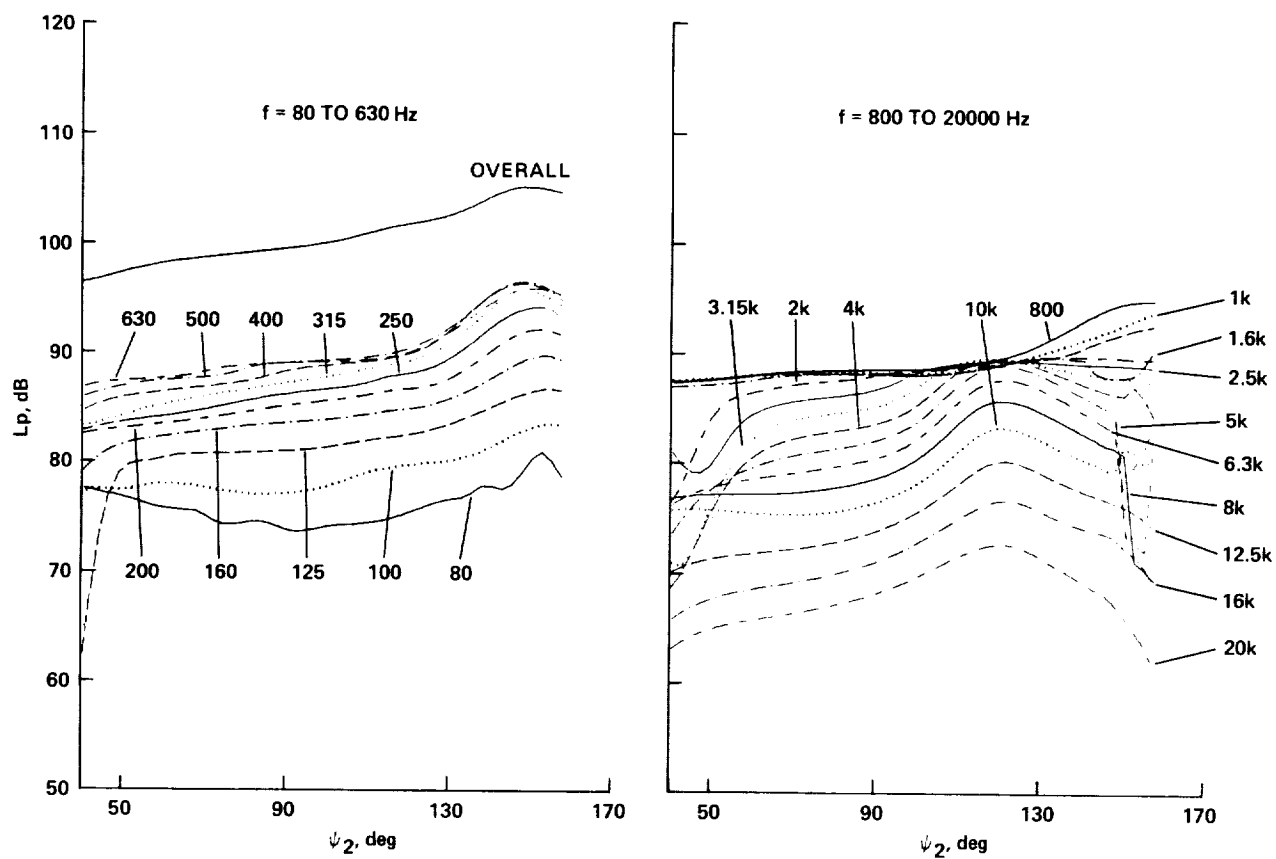
(f) $V_a = 88 \text{ m/sec}$, $V_j = 577 \text{ m/sec}$; 122-m sideline.

Figure 19. Continued.



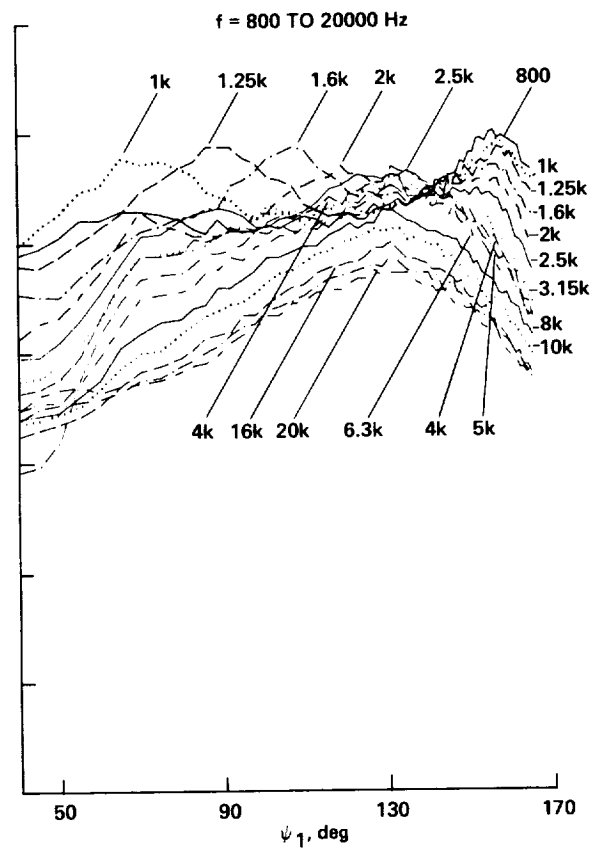
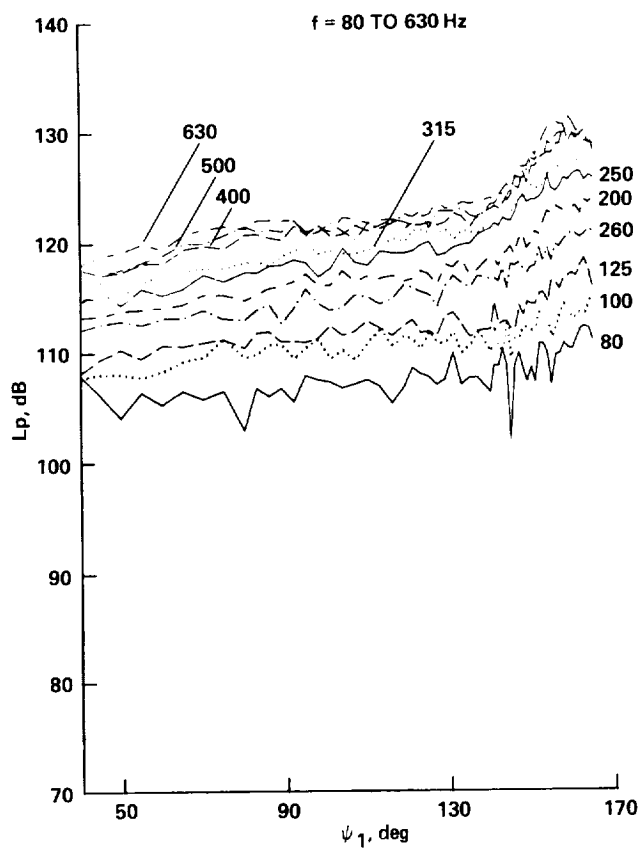
(g) $V_a = 113 \text{ m/sec}$, $V_j = 597 \text{ m/sec}$; 2.0-m sideline.

Figure 19. Continued.



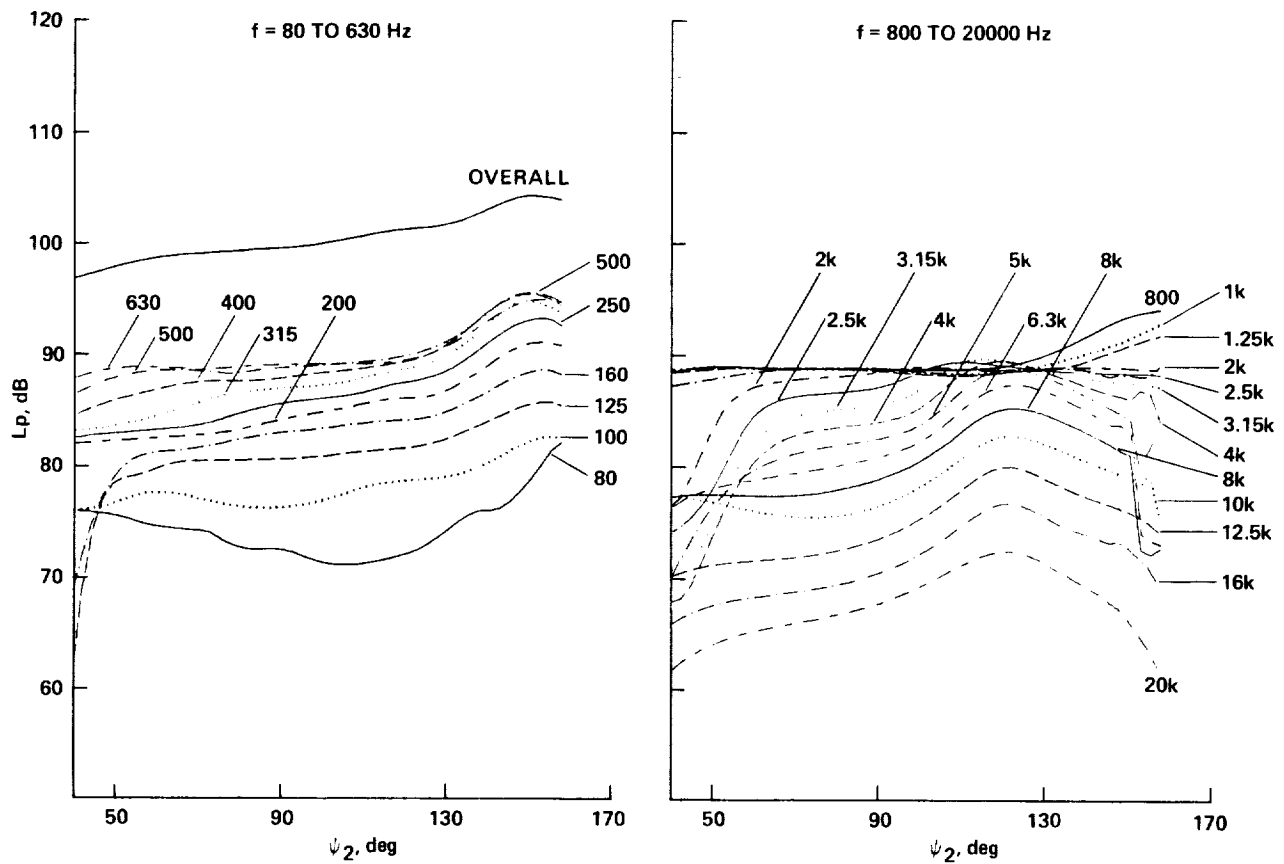
(h) $V_a = 113 \text{ m/sec}$, $V_j = 597 \text{ m/sec}$; 122-m sideline.

Figure 19. Continued.



(i) $V_a = 130$ m/sec, $V_j = 607$ mm/sec; 2.0-m sideline.

Figure 19. Continued.



(j) $V_a = 130$ m/sec, $V_j = 607$ m/sec; 122-m sideline.

Figure 19. Concluded.

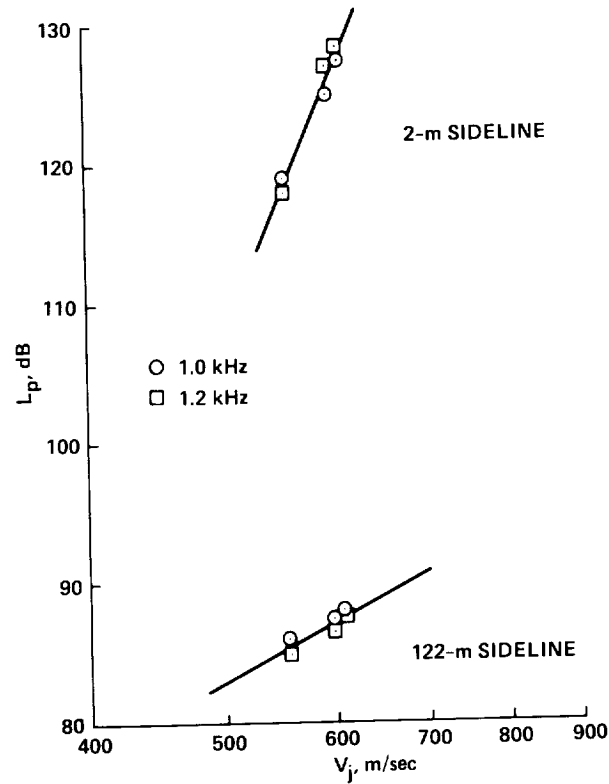


Figure 20. Variation of near field internal or shock noise (upper curve) and far field jet/internal noise (lower curve) versus jet speed from figs. 19(c)–19(j) at two frequencies where internal noise is strong. The data are peak levels between 60° and 90° radiation angles.

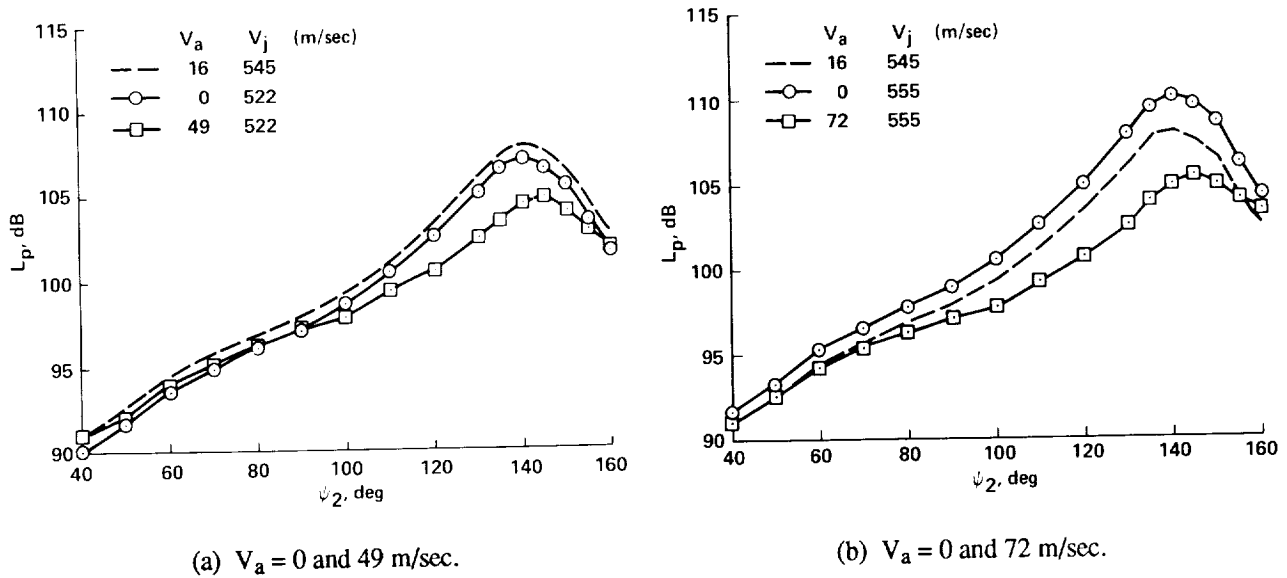
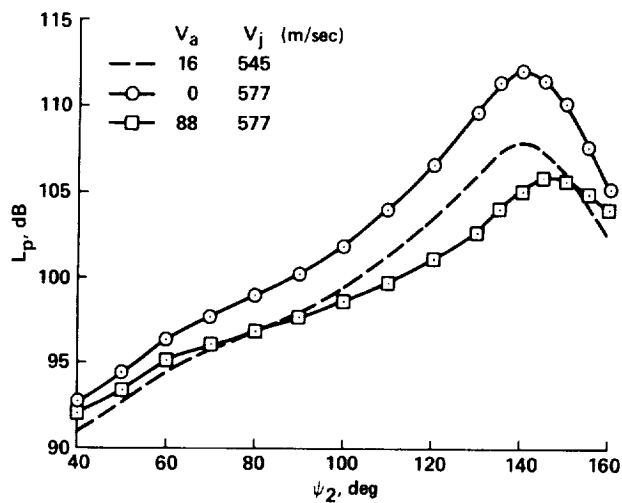
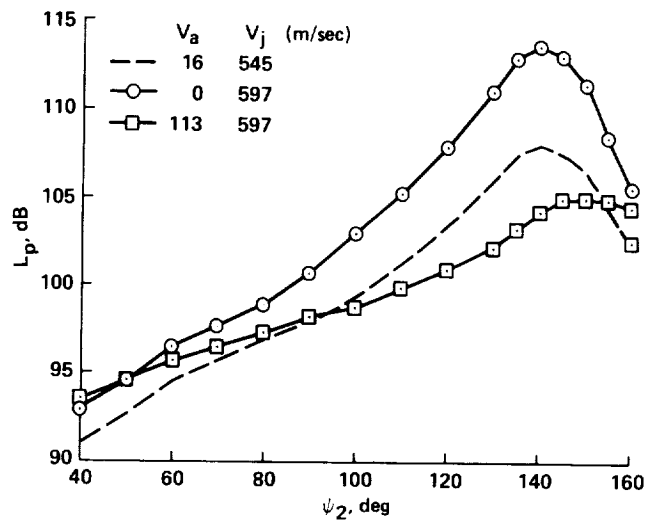


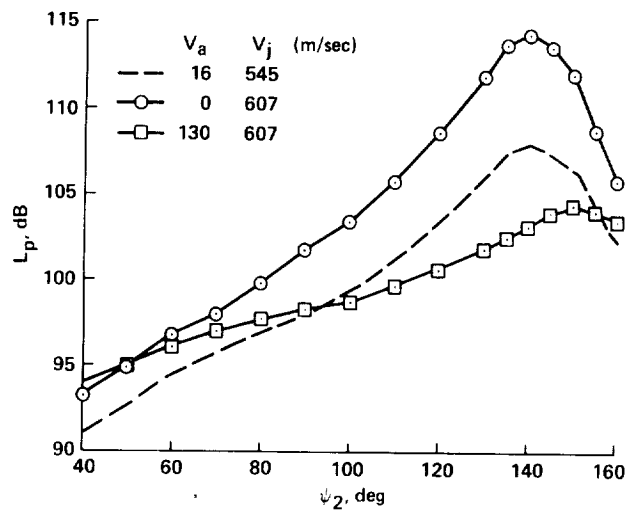
Figure 21. Comparisons of J-85 static and S1 simulated-flight overall sound levels at 122-m sideline. All data corrected for S1 amplification listed in table 1. The nominally static data (dashed curve) were corrected to zero windspeed and the proper jet speed (circles).



(c) $V_a = 0$ and 88 m/sec.

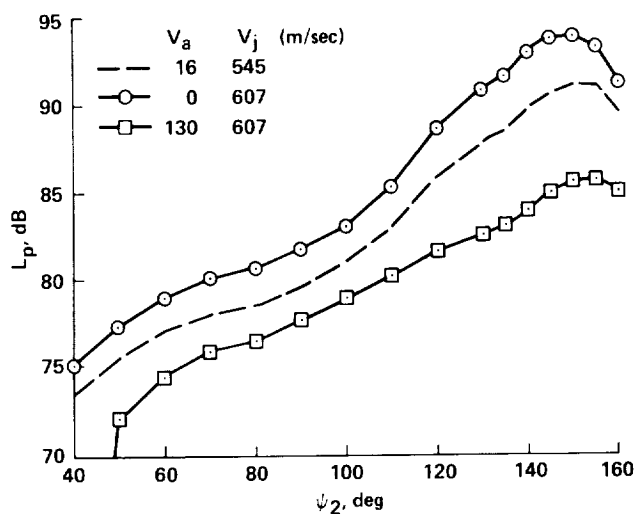


(d) $V_a = 0$ and 113 m/sec.

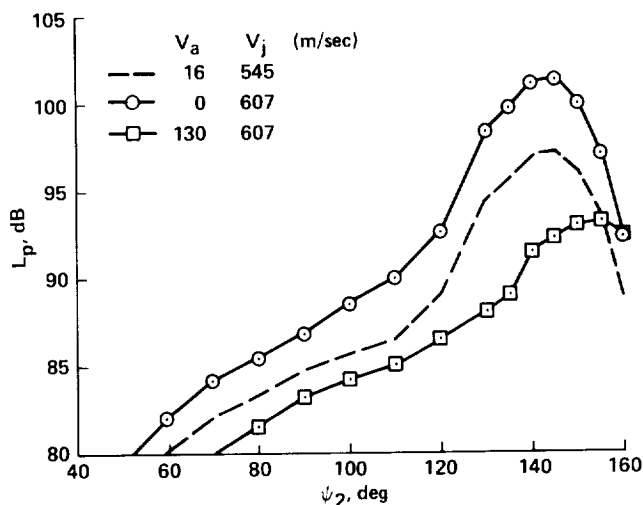


(e) $V_a = 0$ and 130 m/sec.

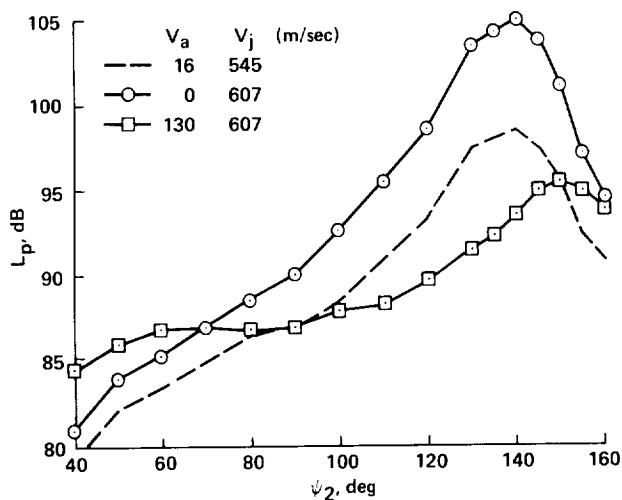
Figure 21. Concluded.



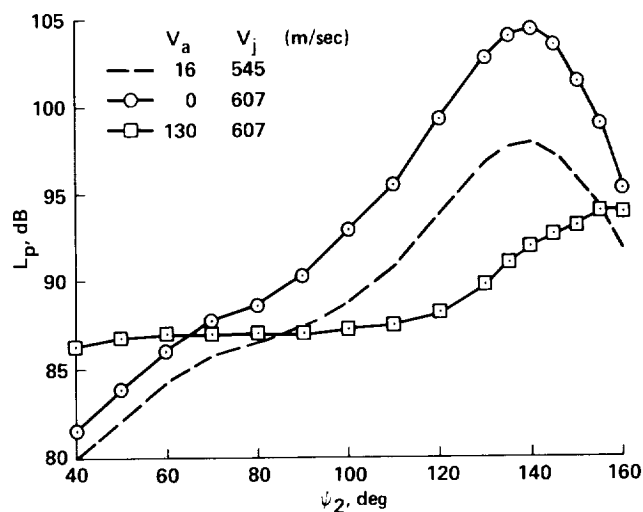
(a) $f = 125$ Hz.



(b) $f = 250$ Hz.

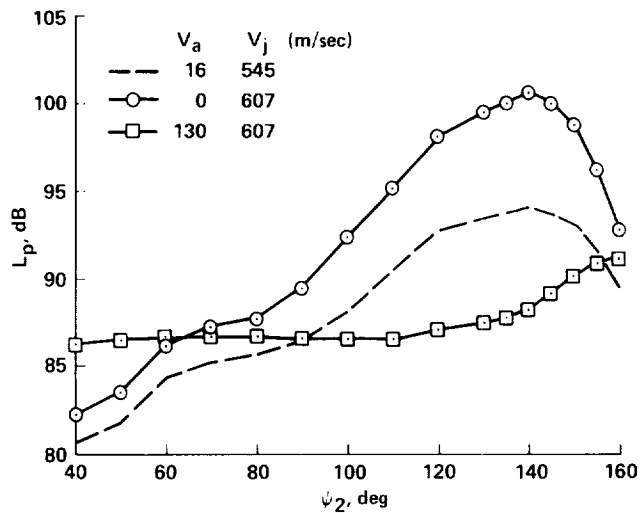


(c) $f = 500$ Hz.

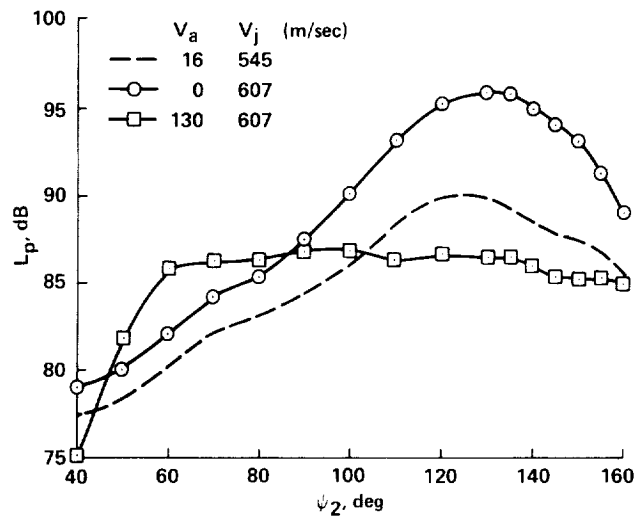


(d) $f = 800$ Hz.

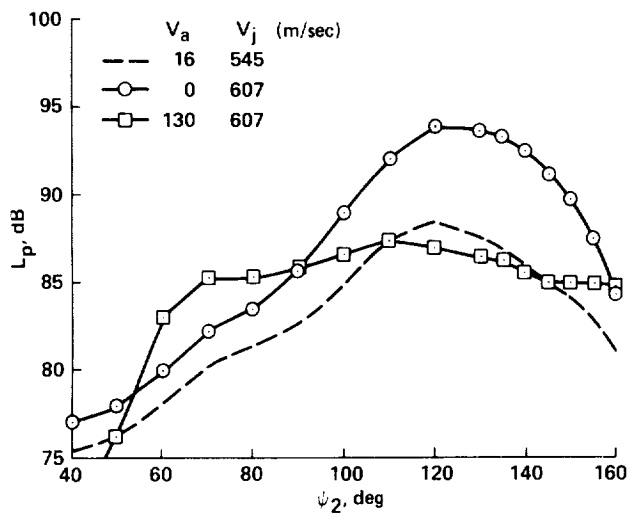
Figure 22. Comparisons of J-85 static and S1 simulated flight noise in third-octave bands at 122-m sideline; $V_a = 0$ and 130 m/sec. All data corrected for S1 amplification listed in table 1. The nominally static data (dashed curve) were corrected to zero windspeed and the proper jet speed (circles).



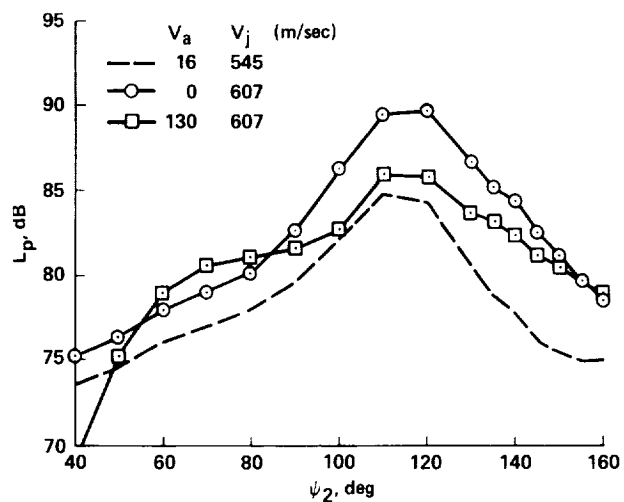
(e) $f = 1250$ Hz.



(f) $f = 2000$ Hz.

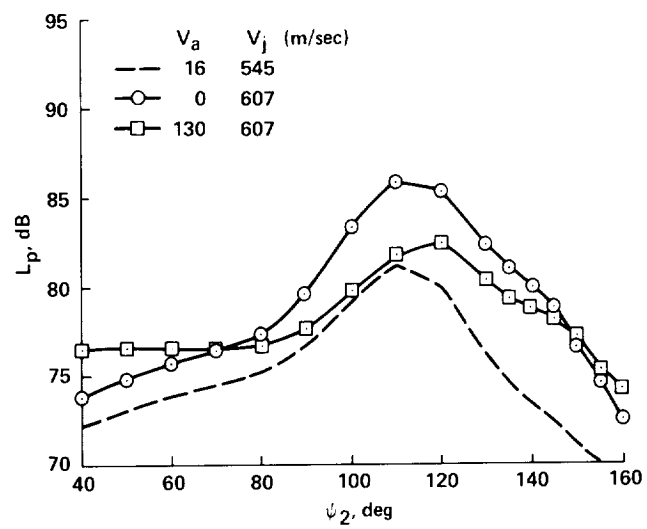


(g) $f = 2500$ Hz.



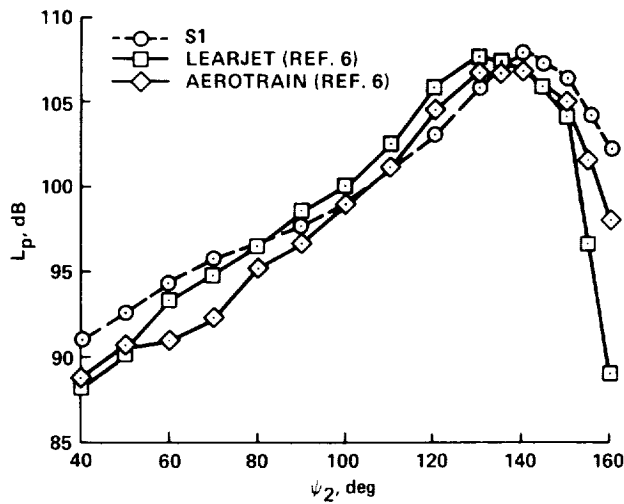
(h) $f = 5000$ Hz.

Figure 22. Continued.

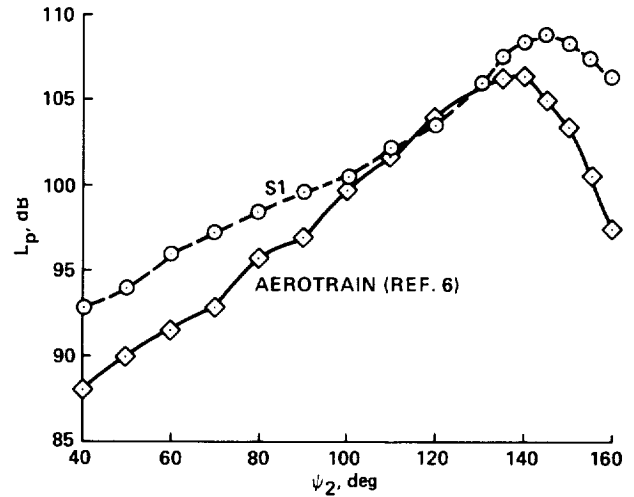


(i) $f = 8000$ Hz.

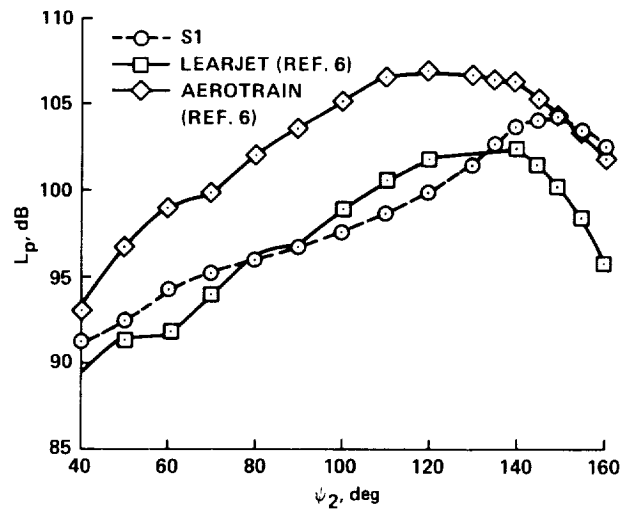
Figure 22. Concluded.



(a) $V_a = 0$, $V_j = 545$ m/sec.



(b) $V_a = 43$ m/sec, $V_j = 558$ m/sec.



(c) $V_a = 82$ m/sec, $V_j = 558$ m/sec.

Figure 23. Comparisons of S1 Wind Tunnel, flyover, and moving train overall sound levels of J-85 at 122-m sideline. S1 data corrected for wind tunnel amplification (table 1). Aerotrain and Lear Jet data corrected to S1 windspeed and J-85 jet speed using equation (5).

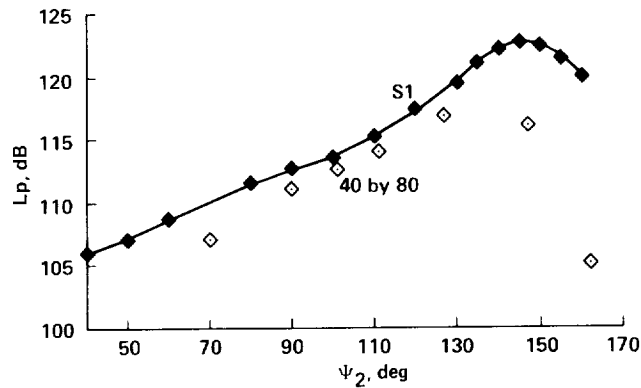
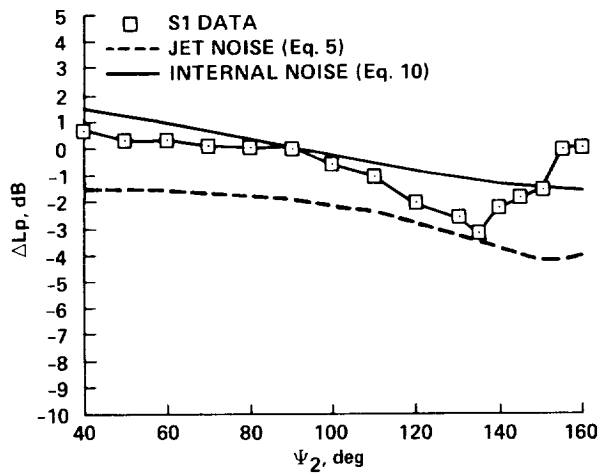
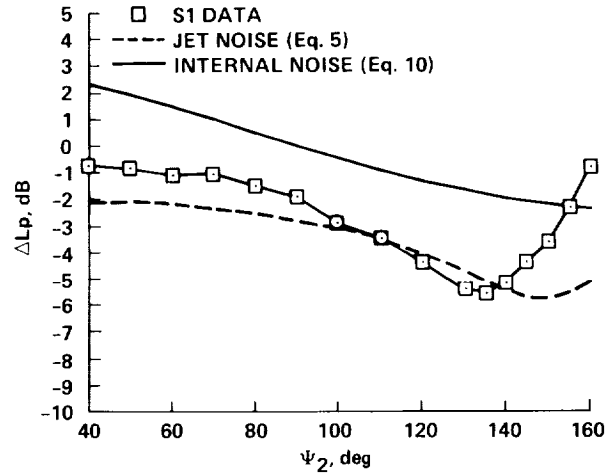


Figure 24. J-85 overall sound levels as measured in the S1 Wind Tunnel and the Ames 40- by 80-Ft Wind Tunnel ($V_a = 52$ m/sec; $V_j = 582$ m/sec). The S1 data were corrected to the 40 by 80 wind and jet speed, and then extrapolated from 122 m to 30.5 m. The 40 by 80 data (ref. 10) were corrected for near field effects and wind tunnel reverberations, and extrapolated from 4.3 m to 30.5 m.

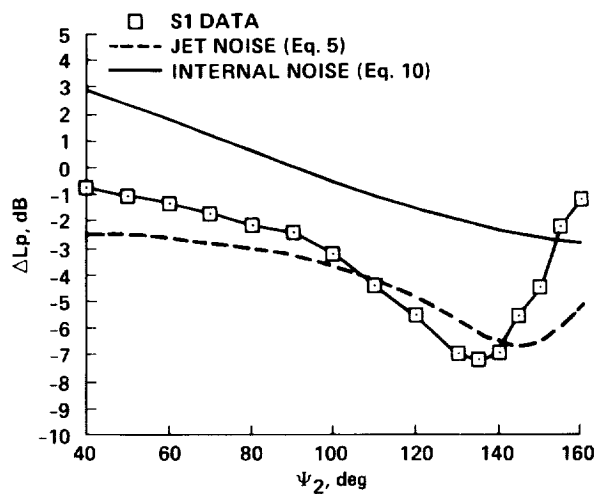


(a) $V_a = 49$ m/sec, $V_j = 522$ m/sec.

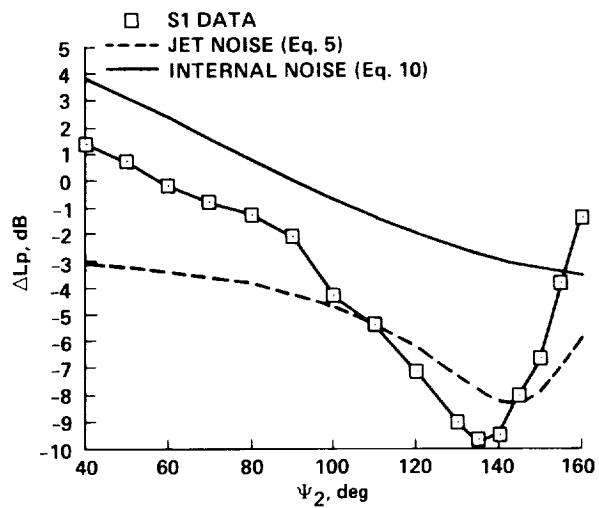


(b) $V_a = 72$ m/sec, $V_j = 555$ m/sec.

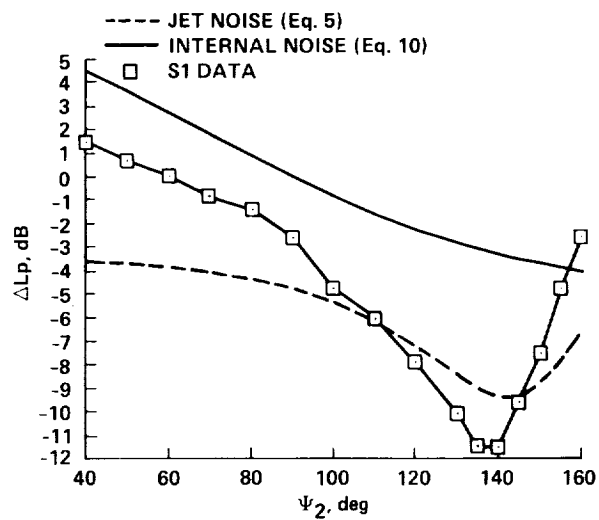
Figure 25. A comparison of measured and predicted flight effects in S1 Wind Tunnel. The difference between measured flight and static noise of figures 21(a)–21(e) is compared with predicted flight effects for pure jet noise (eq. 5) and internal engine noise (eq. 10).



(c) $V_a = 88$ m/sec, $V_j = 577$ m/sec.

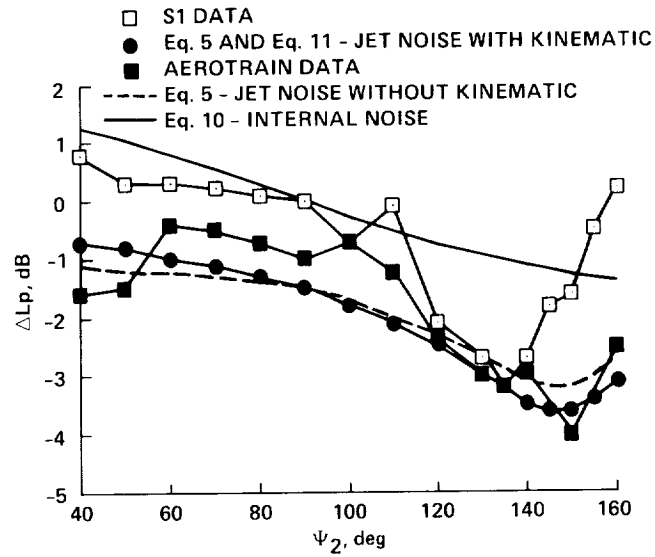


(d) $V_a = 113$ m/sec, $V_j = 597$ m/sec.

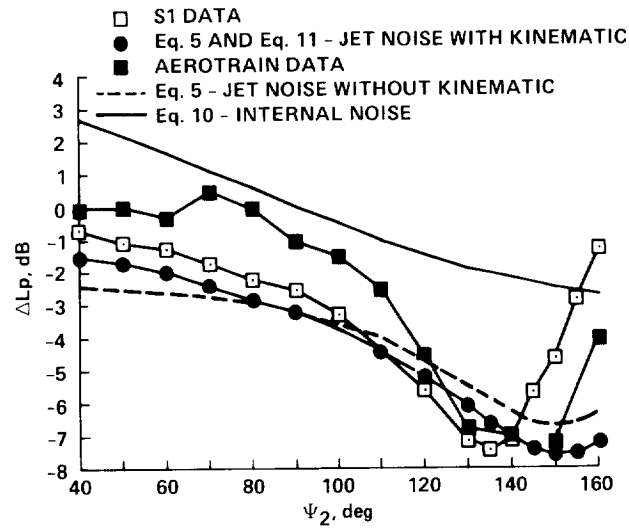


(e) $V_a = 130$ m/sec, $V_j = 607$ m/sec.

Figure 25. Concluded.



(a) $V_a = 41$ m/sec.



(b) $V_a = 82$ m/sec.

Figure 26. S1 and Aerotrain flight effects (difference between forward speed and static J-85 noise) compared with predictions for pure jet and internal noise sources. The pure jet noise predictions were made with and without kinematic effects. $V_j = 546$ m/sec.

APPENDIX A

NOISE 3

```

*
C PROGRAMME SOURCE N03BD
C
C -----
C
C MAIN PROGRAM FOR FIRST PHASE OF ANALYSIS
C
C LOGICAL UNIT ASSIGNMENTS --
C
C      UNIT      FILE
C      05      INPUT FILE FOR ENGINEER SUPPLIED PARAMETERS(IUN)
C      06      OUTPUT FILE (PRINT OUTPUT)
C      07      BINARY OUTPUT FILE
C      08      CORRECTION FILE INPUTS
C      10      SPL DATA FILE(LUN)
C
C -----
C
C*****
      REAL*8  DECB,ANGLES,XDIS,HTMIKE,HTSRCE,SDHUM,SDTEMP,V8,UNITS,
1          THETAM,ADEL,BDEL,AN,YF,YY,ANH1,ANL0,AFHI,AFL0,
2          C0FA,C0FB,TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,VJET,
3          FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C*****
      REAL*8  ABDBM,DISTME,FTOM,PRESS,RTOK,TEMPK
      REAL*8  STR,DFMAX,XPEAKN,XPEAKF,DUMP1(132)
C*****
      REAL*8  JETN(130,33),DUMP(3446)
      COMMON /SUB/ MIKES,MIKEA,MIKEB,ICALL,DECB(130,35),
1          ANGLES(130),XDIS(130,2),HTMIKE(130),HTSRCE,SDHUM,
2          SDTEMP,V8,UNITS,KFIT(2),IWTB,IWTE,
3          THETAM(33),ADEL,BDEL,AN,IGRC,NRUN,
4          YF,YY,ANH1,ANL0,AFHI,AFL0,C0FA(16,35),C0FB(16,35),
5          TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,
6          IBID0,IMRC,IBNC,IRC,IAAC,IDATE,
7          VJET,FREQ(33),THETAS(50,33),XOVERD(50,33),NTAB(33),
8          NTEST,XDMAX(33),THETSM(33),RNZDIA,NZTYPE(10),NFREQ
C
      COMMON /OUTFLG/ IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
C
C*****
      COMMON /TRAFLG/KTRAC,KTDECB,KTDIFF,KTXSD,KTPEAK
C*****
      COMMON /TABD/STR(33),DFMAX(33),XPEAKN(33),XPEAKF(33)
C*****
      COMMON /IDENT/NP01,KMIC,NP012
      COMMON /TITRE/LIB(10),IDAT(3),DB1(8,4),IPI(8)
C*****
      DIMENSION ITIT(20),XM0D(44),LIB1(20),LIB2(20)
C
      EQUIVALENCE (DUMP(1),VJET)
      EQUIVALENCE (DUMP1(1),STR(1))
      DATA PRESS,RTOK,FTOM,D2R /1.0, 0.5555556, 0.3048, 0.017453293/
C
C      K0DT 0  TRAITEMENT BANDE MODANE
C      K0DT 1  TRAITEMENT DONNEES AMES SUR CARTES
C      K0DT 2  TRAITEMENT DONNEES AMES SUR BANDES
      READ(5,101)K0DT
101  FORMAT(I5)
C***** MISE A ZERO DU TABLEAU DECB

```

```

      DO 10 J=1,130
      DO 10 I=1,35
      DECB(J,I)=0.
10 CONTINUE
C*****
      R2D = 1.0/D2R
C      LECTURE BANDE ECRITURE DISQUE
      ND=2
      CALL READIN(KODT,KLISF,ITIT,LIB1,LIB2)
C***** PREPARATION DU TRACE
      IF(KTRAC.EQ.0) GO TO 150
      DO 160 I=1,44
      XM0D(I)=0.
160 CONTINUE
      KPLANC=0
      CALL OPENTR(69,XM0D)
150 CONTINUE
C*****
      CALL OSPL (DECB,MIKES,NFREQ)
      CALL SUBPDB (MIKES,DECB,NFREQ)
      WRITE (6,1001)
1001 FORMAT(1H1,30X,55HSOUND PRESSURE LEVELS READ AS INPUTS FOR THIS EX
      *ECUTION /)
      CALL OUTPUT (XDIS,DECB,ANGLES,FREQ,MIKES,NFREQ,HTMIKE)
C
      IF ( IMRC .NE. 1 ) GO TO 40
      ICALL = 1
      CALL CORECT
40 CONTINUE
      IF ( IBNC .NE. 1 ) GO TO 50
      ICALL = 2
      CALL CORECT
50 CONTINUE
      IF ( IRC .NE. 1 ) GO TO 60
      ICALL = 3
      CALL CORECT
60 CONTINUE
      IF ( IAAC.NE. 1 ) GO TO 80
C
C      APPLY ATMOSPHERIC ABSORPTION CORRECTION TO EACH SPL. USE SUBRT. AIFAB.
C      PRESS = ATMOSPHERES
C      RTOK = DEG R TO DEG K CONVERSION
C      FTOM = FEET TO METERS CONVERSION
C
      TEMPK = TTEMP * RTOK
      DO 200 NF = 1 , NFREQ
      CALL AIFAB(PRESS,TEMPK,SDHUM,FREQ(NF),ABDBM )
      DO 220 NM = 1 , MIKES
      DISTME = XDIS(NM, 2) * FTOM
      DECB(NM,NF) = DECB(NM,NF) + ABDBM*DISTME
220 CONTINUE
200 CONTINUE
C
C      APPLY JET NOISE GROUND REFLECTION CORRECTIONS
C
C
80 IF (IGRC.EQ.0.) GO TO 70
      DO 77 NM = 1,MIKES
      DO 77 NF = 1,NFREQ

```

```

77 JETN(NM,NF) = 0.0
   CALL GRNOIS (XDIS(1,2),HTMIKE,JETN,HTSRCE,TTEMPF,MIKES,FREQ,NFREQ)
   WRITE(6,71)
71  FORMAT(1H1,10X,35HGROUND NOISE REFLECTION CORRECTIONS /)
   CALL OUTCOR (JETN,MIKEA,MIKEB,NFREQ)
   DO 75 NM = 1,MIKES
   DO 75 NF = 1,NFREQ
   DECB(NM,NF) = DECB(NM,NF) - JETN(NM,NF)
75  CONTINUE
70  CONTINUE

C
   IF((IMRC.EQ.0.AND.IBNC.EQ.0.AND.IRC.EQ.0.AND.IAAC.EQ.0)GOTO 90
   CALL OSPL (DECB,MIKES,NFREQ)
   CALL SUBPDB (MIKES,DECB,NFREQ)

C
C
   WRITE(6,1002)
1002 FORMAT(1H1,30X,22HDATA AFTER CORRECTIONS /)
   CALL OUTPUT (XDIS,DECB,ANGLES,FREQ,MIKES,NFREQ,HTMIKE)

C
   90 CONTINUE

C
C***** SAUVEGARDE DE DECB AVANT LISSAGE
   IF(KTDIFF.EQ.0) GO TO 130
   CALL SAVDECB(DECB)
130 CONTINUE

C*****
C***** TRACE DES MESURES
   IF(KTDECB.EQ.0) GO TO 170
   NCAR=17
   CALL TDECB(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,MIKEB,
1 'FIG.      MESURES.',NCAR,XMOD)
170 CONTINUE
   CALL KURVFT(KLISF)
C***** TRACE DES POLYNOMES D INTERPOLATION
   IF(KTDECB.EQ.0) GO TO 180
   NCAR=35
   CALL TDECB(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,MIKEB,
1 'FIG.      POLYNOMES D INTERPOLATION.',NCAR,XMOD)
180 CONTINUE
C***** TRACE DES DIFFERENCES
   IF(KTDIFF.EQ.0) GO TO 140
   NCAR=80
   CALL TDIFF(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,MIKEB,
1 ITIT,NCAR,XMOD)
140 CONTINUE

C*****
   WRITE(6,1003)
1003 FORMAT(1H1,30X,24HDATA AFTER CURVE FITTING /)
   CALL OUTPUT (XDIS,DECB,ANGLES,FREQ,MIKES,NFREQ,HTMIKE)

C
C
C
110 CALL DIRECT
   CALL STAB(NFREQ,FREQ,THETSM,XDMAX)
C***** TRACE X/D=F(THETA-S)
   IF(KTXSD.EQ.0) GO TO 230
   CALL TXSD(KPLANC,XOVERD,THETAS,NTAB,NFREQ,XDMAX,THETSM,
1 RNZDIA,VJET,FREQ,LIB1,LIB2,XMOD)
230 CONTINUE

```

```

C***** TRACE DE THETA-S DU PIC=F(STROUHAL)
      IF(KTPEAK.EQ.0) GO TO 240
      CALL TPEAK(KPLANC,THETSM,STR,NFREQ,RNZDIA,VJET,LIB1,LIB2
1,XMOD)
240 CONTINUE
C*****
C
C
C
C      DUMP COMMON TO DISK FILE AND SAVE THE CORRECTED
C      DATA FOR PHASE 2 ANALYSIS . . .
C      FILE IS WRITTEN IN UNFORMATTED BINARY
C
C      120 CONTINUE
C      ECRITURE SUR BANDE
      WRITE(7)DUMP
      WRITE(7)DUMP1
      WRITE(7)NP01
      WRITE(7) LIB
      WRITE(7) IDAT
      WRITE(7) LIB1
      WRITE(7) LIB2
      REWIND 7
C
C***** FIN DE TRACE
      IF(KTRAC.EQ.0) STOP
      CALL CLOSTR(XMOD)
      WRITE(6,1004) KPLANC
1004 FORMAT(1H1///3X,'NOMBRE DE PLANCHES=',I3////)
C*****
      STOP
      END
*DECK READIN
      SUBROUTINE READIN(KODT,KLISF,ITIT,LIB1,LIB2)
C
C      -----
C
C      SUBROUTINE READIN IS RESPONSIBLE FOR READING AND STORING INTO
C      COMMON THE DATA AND PARAMETERS TO BE USED FOR PROCESSING.
C      AS THE DATA IS READ IT IS ALSO PRINTED TO THE LINE PRINTER
C      RESULTS FILE (UNIT 06).
C
C      ENGINEERS INPUTS AND PARAMETERS ARE READ FROM LOGICAL UNIT
C      ASSIGNED TO VARIABLE LUN, USUALLY 5.
C
C      ANGLES AND DECIBEL READINGS FOR EACH MIKE ARE READ FROM
C      LOGICAL UNIT NUMBER IUN WHICH MAY OR MAY NOT BE THE SAME
C      AS LUN DEPENDING ON THE CONFIGURATION CHOSEN BY THE PROGRAMMER.
C
C      -----
C
C*****
      REAL*8 DECB,ANGLES,XDIS,HTMIKE,HTSRCE,SDHUM,SDTEMP,V8,UNITS,
1      THETAM,ADEL,BDEL,AN,YF,YY,ANHI,ANLO,AFHI,AFL0,
2      COFA,COFB,TR,R2D,D2R,SDTEMP,TTEMP,TTEMPF,VJET,
3      FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C*****
      REAL*8 RFACT,THUM,RNZDIN,HTMIK,ALPHA,H,D
C*****
      COMMON /SUB/ MIKES,MIKEA,MIKEB,ICALL,DECB(130,35),

```

```

1          ANGLES(130),XDIS(130,2),HTMIKE(130),HTSRCE,SDHUM,
2          SDTEMP,V8,UNITS,KFIT(2),IWTB,IWTE,
3          THETAM(33),ADEL,BDEL,AN,IGRC,NRUN,
4          YF,YY,ANHI,ANLO,AFHI,AFL0,C0FA(16,35),C0FB(16,35),
5          TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,
6          IBID0,IMRC,IBNC,IRC,IAAC,IDATE,
7          VJET,FREQ(33),THETAS(50,33),XOVERD(50,33),NTAB(33),
8          NTEST,XDMAX(33),THETSM(33),RNZDIA,NZTYPE(10),NFREQ

C          COMMON /OUTFLG/ IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
C
C*****
          COMMON /TRAFLO/KTRAC,KTDECB,KTDIFF,KTXSD,KTPEAK
          COMMON /IDENT/NP01,KMIC,NP012
          DIMENSION ITIT(1),LIB1(1),LIB2(1)
C*****
          DATA IUN,LUN /5,5/
          DATA RFACT /459.6/

C          WRITE(6,800)
          READ(IUN,910)NP01,KMIC,NP012
910  FORMAT(15,1X,11,15)
          READ (IUN,901) NTEST,NRUN,NZTYPE
          WRITE (6,801)  NZTYPE,NTEST,NRUN,NP01,KMIC,NP012

C          READ (IUN,902) IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
          WRITE (6,802)  IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY

C          READ (IUN,902)  IMRC,IBNC,IRC,IAAC,IGRC,KLISF
          WRITE (6,807)  IMRC,IBNC,IRC,IAAC,IGRC,KLISF

C
C***** LECTURE DES FLAGS DE TRACE
          READ(IUN,902) KTRAC,KTDECB,KTDIFF,KTXSD,KTPEAK
          READ(IUN,1910)(ITIT(1),I=1,20)
          READ(IUN,1910)(LIB1(1),I=1,20)
          READ(IUN,1910)(LIB2(1),I=1,20)
          WRITE(6,810)KTRAC,KTDECB,KTDIFF,KTXSD,KTPEAK
C*****
          READ (IUN,903) MIKEA,MIKEB,MIKES,KFIT(1),KFIT(2),IWTB,IWTE

C          READ (IUN,904) AN,YF,YY,ANHI,ANLO,AFHI,AFL0,ADEL,BDEL
          AN=AN*3.2808
          YF=YF*3.2808
          YY=YY*3.2808
          WRITE (6,804)  AN,YF,YY,ANLO,AFL0,ANHI,AFHI,ADEL,BDEL

C          READ (IUN,902) ITMPTP

C          IF (ITMPTP.NE.0.AND.ITMPTP.NE.1) STOP 3
          IF (ITMPTP.NE.1) GOTO 200
          READ (IUN,905) V8,THUM,TTEMPF,SDHUM,SDTEMF,RNZDIN
          TTEMP = TTEMPF+RFACT
          SDTEMP = SDTEMF+RFACT
          GOTO 210
200  READ (IUN,905) V8,THUM,TTEMP,SDHUM,SDTEMP,RNZDIN
          TTEMPF = TTEMP-RFACT
          SDTEMF = SDTEMP-RFACT
C          RNZDIN IS IN METRES, SWITCH TO RNZDIA IN FEET FOR
C          CALCULATIONS TO BE DONE IN SUBROUTINE DIRECT
210  RNZDIA = RNZDIN*3.2808

```



```

WRITE (6,806) V8,THUM,SDHUM,RNZDIN,RNZDIA,TTEMPF,TTEMP,
*          SDTEMF,SDTEMP
C
C
C   THESE VARIABLES ARE SET EQUAL DUE TO NUMEROUS PROGRAM CHANGES.
C
C   VJET = V8
C
C   READ (IUN,908) HTSRCE,HTMIK
C
C   IF(KODT.NE.0)GO TO 100
C   LECTURE 2 LIGNES DE MICROS
C   RECHERCHE DES ANGLES DE LA LIGNE 2 EN TOTALITE
C   RECHERCHE DES ANGLES DE LA LIGNE 1 EGAUX A CEUX DE LA LIGNE 2
C   CALL RECHPOIN(NFREQ,MIKES,MIKEA,MIKEB,DECB,FREQ,ANGLES)
C   GO TO 300
100 IF(KODT.EQ.1)CALL LCART
   IF(KODT.EQ.2)CALL RECHNASA(NFREQ,MIKES,MIKEA,MIKEB,DECB,FREQ,ANGLE
   *S)
300 WRITE (6,803) MIKEA,MIKEB,MIKES,KFIT(1),KFIT(2),IWTB,IWTE
   IF (MIKES.NE.(MIKEA+MIKEB)) STOP 1
   DO 101 J=1,MIKES
101  HTMIKE(J)=HTMIK
   WRITE (6,809) HTSRCE,(HTMIKE(I),I=1,MIKES)
809  FORMAT( / 1X,20HSOURCE HEIGHT (FT) =, F6.2 /
   * 1X,20HMICROPHONE HEIGHT =, 15F6.2 / 2(21X,15F6.2/) 21X,5F6.2)
C
C
C   CALCULATION OF MIKE DISTANCES BASED ON
C   MIKE HEIGHT AND ANGLE ... SEE PROGRAM DOCUMENTATION
C   FOR REFERENCE ON METHOD USED (GEOMETRY)
C
C   DO 110 L=1,MIKES
C   IF (L.GT.MIKEA) GOTO 111
C   IF (HTSRCE.EQ.HTMIKE(L)) GOTO 112
C   XDIS(L,2) = AN/DSIN((180.-ANGLES(L))*D2R)
C   XDIS(L,1) = XDIS(L,2)*DCOS((180.-ANGLES(L))*D2R)
C   GOTO 110
112 IF (ANGLES(L).GT.90.) ALPHA=90.-ANGLES(L)
   IF (ANGLES(L).LT.90.) ALPHA=ANGLES(L)-90.
   H = HTSRCE-HTMIKE(L)
   IF (ANGLES(L).EQ.90.) GOTO 113
   D = AN/DCOS(ALPHA*D2R)
   XDIS(L,1) = DSQRT(AN*AN+D*D)
   XDIS(L,2) = DSQRT(D*D+H*H)
   GOTO 110
113 XDIS(L,1) = 0.0
   XDIS(L,2) = DSQRT(AN**2+H**2)
   GOTO 110
111 IF (HTSRCE.EQ.HTMIKE(L)) GOTO 114
   XDIS(L,2) = YF/DSIN((180.-ANGLES(L))*D2R)
   XDIS(L,1) = XDIS(L,2)*DCOS((180.-ANGLES(L))*D2R)
   GOTO 110
114 IF (ANGLES(L).GT.90.) ALPHA=90.-ANGLES(L)
   IF (ANGLES(L).LT.90.) ALPHA=ANGLES(L)-90.
   H = HTSRCE-HTMIKE(L)
   IF (ANGLES(L).EQ.90.) GOTO 115
   D = YF/DCOS(ALPHA*D2R)
   XDIS(L,1) = DSQRT(YF**2+D**2)
   XDIS(L,2) = DSQRT(D*D+H*H)

```

```

      GOTO 110
115 XDIS(L,1) = 0.0
      XDIS(L,2) = DSQRT(YF**2+H**2)
110 CONTINUE
      WRITE (6,805) (XDIS(L,1),L=1,MIKES)
      WRITE (6,808) (XDIS(L,2),L=1,MIKES)
C
C*****INPUT FORMATS
C
901 FORMAT (I3,I2,10A4)
902 FORMAT (6I1)
903 FORMAT (7I2)
904 FORMAT (10F6.2)
905 FORMAT (5F10.1,F10.2)
908 FORMAT (16F5.0)
909 FORMAT (16F5.2)
1910 FORMAT(20A4)
C
C*****OUTPUT FORMATS
C
800 FORMAT(1H1,37HSOURCE LOCATION PROGRAM - STATIC CASE // 1X,
      * 40HINPUT PARAMETERS READ FOR THIS EXECUTION /)
801 FORMAT(1X,16HIDENTIFICATION =,1X, 10A4,5X,4HTEST,14,5H RUN ,
      * I3,30X,14,4H M ,I1,I5)
802 FORMAT(1X,16HOUTPUT FLAGS =,1X,6I1)
803 FORMAT(/ 1X,19HNEAR FIELD MIKES = ,12,10X,
      *18HFAR FIELD MIKES = ,12,10X,14HTOTAL MIKES = ,13 /
      *1X,10HKFIT(1) = ,12,19X,10HKFIT(2) = ,12,/1X,
      *10HIWTB = ,12,19X,10HIWTE = ,12)
804 FORMAT(/ 12X,10HNEAR FIELD, 6X,9HFAR FIELD /
      *1X,8HDISTANCE,F11.3,5X,F11.3,5X,F11.3 / 1X,3HLOW,
      *F16.3,5X,F11.3 / 1X,2HHI,F17.3,5X,F11.3,5X,5HDELTA,F14.3,5X,F11.3)
806 FORMAT(/ 1X,8HV(INF) =,F8.2 /
      * 1X,8HTHUM =,F8.2,10X,7HSDHUM =,F8.2,10X,12HNOZZLE DIA =,
      *F8.4,9H METRES =,F8.4,5H FEET,
      * // 21X,5HDEG F,10X,5HDEG R / 1X,11HTUNNEL TEMP,7X,
      *F7.1,8X,F7.1 / 1X,12HSTD DAY TEMP,6X,F7.1,8X,F7.1)
805 FORMAT(/ 1X,14HMIKE DISTANCES /
      *1X,13HCENTER LINE =,10F10.3,/ 3(14X,10F10.3 /))
807 FORMAT(1X,19HCORRECTION FLAGS =,1X,6I1)
808 FORMAT(/ 1X,13HSOURCE DIST =,10F10.3 / 3(14X,10F10.3 /))
810 FORMAT(1X,'PLOT FLAGS =',5I1)
      END
*DECK OSPL
      SUBROUTINE OSPL (DECB,MIKES,NFREQ)
C*****
      REAL*8 DECB(130,35)
      REAL*8 D,DDB,S
C*****
      DO 400 II=1,MIKES
      S = DECB(II,1)
      DO 390 L=2,NFREQ
      D=DECB(II,L)
      IF (D.LT.40.) GO TO 390
      DDB=DABS(S-D)
      IF (DDB.GT.7.5) GO TO 380
      S=DEXP(1.1115)*DEXP(-.19077*DDB)+DMAX1(S,D)
      GO TO 390
380 S=DEXP(1.1406)*DEXP(-.20172*DDB)+DMAX1(S,D)
390 CONTINUE

```

```

      DECB(11,34) = S
400 CONTINUE
      RETURN
      END
*DECK PNDB
      SUBROUTINE PNDB(LP,NF,PLDB,DL,ALG,ANN,ANG)
C
C      THIS SUBROUTINE HAS BEEN MODIFIED TO WORK ONLY FOR ONE-THIRD
C      OCTAVE BAND WIDTH.
C
C*****
      REAL*8 SUMN,PLDB
C*****
      REAL*8 DL(24,2),ALG(24,2),ANN(24,2),ANG(24,2)
      REAL*8 LP(27),LB(24),NOY(24),NMAX,NBAR
      N=NF
      IF(N.GT.24)N=24
      DO 21 I=1,24
21 LB(I)=LP(I)
      NMAX=0.0
      SUMN=0.0
      DO 13 I=1,N
      NOY(I)=DMIN1(ANG(I,1)*ANN(I,1)**((LB(I)-ALG(I,1))/DL(I,1)),
1          ANG(I,2)*ANN(I,2)**((LB(I)-ALG(I,2))/DL(I,2)))
      IF(NOY(I).GE.NMAX)NMAX=NOY(I)
13 SUMN=SUMN+NOY(I)
      NBAR = NMAX+0.15*(SUMN-NMAX)
      PLDB=40.0+10.0*DLOG10(NBAR)/DLOG10(2.0)
      IF(PLDB.LT.0.0)PLDB=0.0
      RETURN
      END
*DECK SUBPDB
      SUBROUTINE SUBPDB (MIKES,DECIBL,NFREQ)
      REAL*8 PDB
      REAL*8 DECIBL(130,35),LP(27)
      REAL*8 DL(24,2),ALG(24,2),ANN(24,2),ANG(24,2)
      DATA DL/15*10.,9*110.,30.,25.,2*26.
1          ,28.,2*27.,30.,51.,6*10.,7*110.,6.,9./
      DATA ALG/52.,51.,49.,47.,46.,45.,43.,42.,41.,5*40.,38.,34.
1          ,32.,30.,2*29.,30.,31.,34.,37.,64.,60.,56.,53.,51.,48.
2          ,46.,44.,42.,5*40.,38.,34.,32.,30.,2*29.,30.,31.,37.,41./
      DATA ANN/15*2.,9*1975.,13.5,10.3,2*9.07,9.76,2*7.94,9.15,
1          136.7,6*2.,7*1975.,1.79,2.4/
      DATA ANG/48*1.0/
C
      L24 = 24
      IF ( NFREQ .LT. 24 ) L24 = NFREQ
      DO 280 I1 = 1,MIKES
      KSW=0
      DO 260 L=1,L24
      LP(L) = DECIBL(I1,L+6)
      IF (LP(L).GT.0.0) KSW = 1
      IF (LP(L).LT.0.0) LP(L) = 0.0
260 CONTINUE
      IF (KSW.GT.0) GO TO 270
      PDB=0.
      GO TO 280
270 CONTINUE
      NF = L24
      CALL PNDB(LP,NF,PDB,DL,ALG,ANN,ANG)

```

```

C
C      IF ( DABS( YM1 ) .LT. EPS ) GO TO 190
C
C      KNTBI = 0
C
C      X1 = XM1
C      X2 = XM2
C
C*****
C
C
C      USE FOR DEBUGGING .
C
C      IF(IPRINT.EQ.1) WRITE(6,97) X1,X2,YM1,YM2
C      97 FORMAT(2X,26HSTART WITH X1,X2,YM1,YM2 , 4E16.8 )
C
C*****
C
C      99 CONTINUE
C
C      KNTBI = KNTBI + 1
C      IF ( KNTBI .GE.ITER ) GO TO 1500
C      DIVIDE THE INTERVAL , (X1,X2) AND CHECK BOTH HALVES FOR THE ROOT.
C      THE TWO HALVES ARE (X1,XBAR) , (XBAR,X2)
C
C      XBAR = ( X1 + X2 ) / 2.
C      IF ( DABS(X2-X1) .LE. EPS ) GO TO 150
C      120 CONTINUE
C      Y1 = POLYX ( COEF,N,X1 )
C      YBAR = POLYX ( COEF,N,XBAR )
C
C      CHECK FOR THE ROOT IN THE 1ST HALF .
C
C      CHECK = Y1*YBAR
C      IF ( CHECK .LE. 0.0 ) X2 = XBAR
C      IF ( CHECK .LE. 0.0 ) GO TO 99
C
C
C      CHECK FOR THE ROOT IN THE 2ND HALF .
C
C      X1 = XBAR
C      GO TO 99
C
C      150 CONTINUE
C
C      YY = POLYX ( COEF,N,XBAR )
C      IF ( DABS(YY) .GT. EPS ) GO TO 120
C      KNTR = KNTR + 1
C      ROOTS(KNTR) = XBAR
C      YROOTS(KNTR) = YY
C      XM2 = XBAR
C      YM2 = 0.0
C
C      IF(IPRINT.EQ.1)WRITE(6,170) KNTR,KNTBI
C      170 FORMAT ( 1X, 9HROOT NUM. , 13,5H USED , 13,11H ITERATIONS )
C
C
C
C      GO TO 190
C      1500 CONTINUE

```

```

      IERR = 1
      IF ( IPRINT .NE. 1 ) GO TO 190
      WRITE ( 6,1600)ITER, EPS
1600  FORMAT (1X,12,45H ITERATIONS FAILED TO GIVE DESIRED ACCURACY( , -
      2E14.7,2H ) / )
      WRITE(6,1700) X1,X2
1700  FORMAT ( 1X, 7HX1,X2 = , 2E18.9 / )
      190 CONTINUE
      XM1 = XM2
      YM1 = YM2
C
C
      200 CONTINUE
C
      250 CONTINUE
C
      IF ( KNTR .LT. 1 ) GO TO 600
      GO TO 900
C
      300 CONTINUE
C
C
C
      IF N=2 , WE HAVE THE 2ND DEGREE EQ. , QUADRATIC FORMULA WILL BE USED.
C
      IF ( N .LT. 2 ) GO TO 400
      DISCRM = COEF(N)**2 - 4. * COEF(N+1) * COEF(1)
      IF ( DISCRM .LT. 0.0 ) GO TO 600
      ROOTS(1) = (-COEF(N) + DSQRT(DISCNM)) / ( 2.*COEF(N+1) )
      ROOTS(2) = (-COEF(N) - DSQRT(DISCNM)) / ( 2.*COEF(N+1) )
      KNTR = 2
C
      YROOTS(1) = POLYX(COEF,N,ROOTS(1) )
      YROOTS(2) = POLYX(COEF,N,ROOTS(2) )
C
      GO TO 900
C
      400 CONTINUE
C
      KNTR = 1
      ROOTS(1) = -COEF(1)/COEF(2)
      YROOTS(1) = POLYX ( COEF,N,ROOTS(1) )
      GO TO 900
C
C
      600 CONTINUE
C
      IF(IPRINT.EQ.1)WRITE(6,650) N
      650  FORMAT ( 1X,14HNO REAL ROOTS. , 5X,13,19H DEGREE POLYNOMIAL. / )
      IERR = 1
      GO TO 2000
C
      900 CONTINUE
C
      IF ( IPRINT .NE. 1 ) GO TO 2000
C
      WRITE ( 6,920) N
      920  FORMAT ( 1X,15HROOT(S) OF THE , 13, 18H DEGREE POLYNOMIAL )
C
      IF ( KNTR .GT. 8 ) GO TO 1000
C

```

```

        WRITE ( 6,950) ( ROOTS(I) , I = 1 , KNTR )
950  FORMAT ( 1X,8E16.8 )
        WRITE ( 6,960 )
960  FORMAT ( 1X,16HTHE Y-VALUES ARE )
        WRITE(6,950) (YROOTS(I),I = 1,KNTR )
        GO TO 2000
980  FORMAT ( 1X,/ )
C
1000 CONTINUE
C
C
        KNTR1 = KNTR/2
        KNTR2 = KNTR - KNTR1
C
        WRITE ( 6,950) ( ROOTS(I) , I = 1 , KNTR1)
        KNTR1 = KNTR1 + 1
        WRITE ( 6,950) ( ROOTS(I) , I = KNTR1 , KNTR2 )
C
        WRITE(6,980)
        KNTR1 = KNTR1 - 1
        WRITE ( 6,960 )
        WRITE(6,950) (YROOTS(I),I = 1,KNTR1)
        KNTR1 = KNTR1 + 1
        WRITE(6,950) (YROOTS(I),I =KNTR1 ,KNTR2)
C
C
2000 CONTINUE
C
        WRITE( 6,25 )
        NROOTS = KNTR
        RETURN
        END
*DECK OUTPUT
        SUBROUTINE OUTPUT (XDIS,DECB,ANGLES,FREQ,MIKES,NFREQ,HTMIKE)
C
C*****
        REAL*8  XDIS(130,2),DECB(130,35),ANGLES(130),FREQ(33)
        REAL*8  HTMIKE(130)
C
        LC = MIKES / 15
        IF(MOD(MIKES,15).NE.0) LC = LC + 1
        IST = 1
        DO 375 J = 1,LC
            ISTP = IST + 14
            IF (ISTP.GT.MIKES) ISTP=MIKES
C
            WRITE (6,610) ((K),K=IST,ISTP)
            WRITE (6,620) (ANGLES(K),K=IST,ISTP)
            WRITE (6,625) (HTMIKE(K),K=IST,ISTP)
            WRITE (6,630) (XDIS(K,1),K=IST,ISTP)
            WRITE (6,640) (XDIS(K,2),K=IST,ISTP)
            WRITE (6,650)
            DO 350 L=1,NFREQ
350  WRITE(6,670) FREQ(L), (DECB(K,L),K=IST,ISTP)
            WRITE(6,710) (DECB(K,34),K=IST,ISTP)
            WRITE(6,720) (DECB(K,35),K=IST,ISTP)
C
            WRITE (6,730)
            IST = ISTP + 1
375  CONTINUE

```

```

C
610 FORMAT (12H MICROPHONE: ,14X,15(1X,13,2X))
620 FORMAT (12H ANGLE(DEG): ,14X,15F6.1)
625 FORMAT (13H HEIGHT (FT): ,13X,15F6.1)
630 FORMAT (14H CL DIST(FT): ,12X,15F6.1)
640 FORMAT (14H REF DIST(FT): ,12X,15F6.1)
650 FORMAT (12H FREQ(HERTZ))
670 FORMAT (2X,F9.0,15X,15F6.1)
710 FORMAT (12H OVERALL SPL,14X,15F6.1)
720 FORMAT (5H PNDB,21X,15F6.1)
730 FORMAT (1H1)
      RETURN
      END
*      DECK LCART
      SUBROUTINE LCART
C*****
      REAL*8 DECB,ANGLES,XDIS,HTMIKE,HTSRCE,SDHUM,SDTEMP,V8,UNITS,
1          THETAM,ADEL,BDEL,AN,YF,YY,ANH1,ANL0,AFHI,AFL0,
2          C0FA,C0FB,TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,VJET,
3          FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C*****
      IUN=LUN=5
      COMMON /SUB/ MIKES,MIKEA,MIKEB,ICALL,DECB(130,35),
1          ANGLES(130),XDIS(130,2),HTMIKE(130),HTSRCE,SDHUM,
2          SDTEMP,V8,UNITS,KFIT(2),IWTB,IWTE,
3          THETAM(33),ADEL,BDEL,AN,IGRC,NRUN,
4          YF,YY,ANH1,ANL0,AFHI,AFL0,C0FA(16,35),C0FB(16,35),
5          TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,
6          IBID0,IMRC,IBNC,IRC,IAAC,IDATE,
7          VJET,FREQ(33),THETAS(50,33),XOVERD(50,33),NTAB(33),
8          NTEST,XDMAX(33),THETSM(33),RNZDIA,NZTYPE(10),NFREQ
      READ (IUN,903) NFREQ
903  FORMAT (7I2)
      READ (IUN,905) (FREQ(I),I=1,NFREQ)
905  FORMAT (8F10.1)
      DO 101 J=1,MIKES
      READ (LUN,909) ANGLES(J), (DECB(J,L),L=1,NFREQ)
909  FORMAT (16F5.2)
101  CONTINUE
      RETURN
      END
*      DECK RECHNASA
      SUBROUTINE RECHNASA(NFREQ,MIKES,MIKEA,MIKEB,DECB,FREQ,ANGLES)
      COMMON/IDENT/IPT1,KBNSA,IPT2
      COMMON/TITRE/LIB(10),IDATE(3),DB1(8,4)
      REAL*8 DECB(130,35),FREQ(33),ANGLES(130)
      DIMENSION ANG(30),SPE(30),IPAR(2),PAR(8),NZTYP(3),FQ(30)
      DIMENSION TAB1(13),TAB2(30),TAB3(31),TAB4(31),TAB5(31)
      EQUIVALENCE (TAB1(1),IPAR(1)),(TAB1(3),PAR(1)),(TAB1(11),NZTYP(1))
      EQUIVALENCE (TAB2(1),FQ(1))
      EQUIVALENCE (TAB3(1),ANG(1))
      EQUIVALENCE (TAB5(1),SPE(1))
      DATA ND1,ND2/2,3/,NDFQ/5/,LIB0/4H
      IF(KBNSA.EQ.1)NFREQ=25
      IF(KBNSA.EQ.2)NFREQ=23
C      LECTURE DU POINT CONTENANT LA LIGNE 1
      READ(ND1,101,END=601)(TAB1(I),I=1,13)
      IF(IPT1.NE.IPAR(1))WRITE(6,104)IPAR(1);STOP
104  FORMAT(' ERREUR POINT LU =',I5)
      NP01=IPAR(1)

```

```

      N=IPAR(2)
      MIKEA=N
101  FORMAT(31A4)
      DB1(1,1)=0.
      DB1(1,2)=PAR(5)/3.2808
      READ(ND1,101)(TAB2(I),I=1,30)
      DO 10 I=1,NFREQ
10   FREQ(I)=FQ(I+NDFQ)
      READ(ND1,101)(TAB3(I),I=1,N)
      DO 20 I=1,N
20   ANGLES(I)=ANG(I)
      READ(ND1,101)(TAB4(I),I=1,N)
      DO 30 IA=1,N
      READ(ND1,101)(TAB5(I),I=1,31)
      DO 40 IDB=1,NFREQ
40   DECB(IA,IDB)=SPE(IDB+NDFQ)
30   CONTINUE
C    LECTURE DU FICHIER CONTENANT LA LIGNE 2
      READ(ND2,101,END=602)(TAB1(I),I=1,13)
      IF(IPT2.NE. IPAR(1))WRITE(6,104)IPAR(1);STOP
      N=IPAR(2)
      MIKEB=N
      DB1(3,1)=0.
      DB1(3,2)=PAR(5)/3.2808
      DO 5 I=1,3
5     LIB(I)=NZTYP(I)
      DO 6 I=4,10
6     LIB(I)=LIB0
      DO 7 I=1,3
7     IDATE(I)=0
      READ(ND2,101)(TAB2(I),I=1,30)
      READ(ND2,101)(TAB3(I),I=1,30)
      DO 50 I=1,N
50   ANGLES(MIKEA+I)=ANG(I)
      READ(ND2,101)(TAB4(I),I=1,30)
      DO 60 IA=1,N
      READ(ND2,101)(TAB5(I),I=1,31)
      DO 70 IDB=1,NFREQ
70   DECB(MIKEA+IA,IDB)=SPE(IDB+NDFQ)
60   CONTINUE
      MIKES=MIKEA+MIKEB
      GO TO 900
601  WRITE(6,102)ND1
102  FORMAT(' FIN DE FICHIER ETIQ. LOGIQUE ',I2)
      STOP
602  WRITE(6,103)ND2
103  FORMAT(' FIN DE FICHIER ETIQ. LOGIQUE ',I2)
      STOP
900  RETURN
      END
      SUBROUTINE RECHPOIN(NFREQ,MIKES,MIKEA,MIKEB,DECB,FREQ,
      *ANGLES)
C
C    INTERFACE PROG AMES ET PROG ONERA
C                                     NO MICROS
C    IPI(1)=0- 7M,LIGNE 1,MICRO 1      1
C    IPI(2)=0- 7M,LIGNE 1,MICRO 2      2
C    IPI(3)=0- 7M,LIGNE 2,MICRO 1      3
C    IPI(4)=0- 7M,LIGNE 2,MICRO 2      4
C    IPI(5)=7-14M,LIGNE 1,MICRO 1      5

```



```

C      IPI(6)=7-14M,LIGNE 1,MICRO 2      6
C      IPI(7)=7-14M,LIGNE 2,MICRO 1      7
C      IPI(8)=7-14M,LIGNE 2,MICRO 2      8
C
C*****
COMMON /IDENT/NP01,KMIC
COMMON /TITRE/LIB(10),IDATE(3),DB1(8,4),IPI(8)
C*****
REAL*8 OMEGA(8,40),DECB(130,35),FREQ(33),ANGLES(130),AMI,AMA
REAL*8 AUX(130,30)
DIMENSION LTT(10),JFR(44),IRG(2)
DIMENSION IDB1(8,6),TETA(8,40),SDB(8,40,50)
DIMENSION TAB1(050),TAB2(050),ITAB(6),TAB(4),ANG(40),SPE(45)
EQUIVALENCE(TAB1(1),TITR)
EQUIVALENCE (TAB1(2),ITAB(1))
EQUIVALENCE (TAB1(8),TAB(1))
EQUIVALENCE (TAB1(12),LTT(1))
EQUIVALENCE (TAB1(22),ANG(1))
EQUIVALENCE(TAB2(1),ANGLE)
EQUIVALENCE(TAB2(2),SPE(1))
C
C      DATA IPI/1,2,3,4,5,6,7,8/
C
C      DATA JFR/3,4,5,6,8,10,12,16,20,25,31,40,50,63,80,100,125,160,200,
*250,315,400,500,630,800,1000,1250,1600,2000,2500,3150,4000,5000,
*6300,8000,10000,12500,16000,20000,25000,31500,40000,50000,63000/
C      DATA IUN/5/
C
C      FREQUENCE 80 A 20000HZ
C
C      ND=2
C      NUMER =1
C      REWIND ND
50 KODR=0
600 CONTINUE
READ(ND,105,END=610)(TAB1(I),I=1,50)
105 FORMAT(50A4)
IF(ITAB(1).NE.NP01 )GO TO 500
C      NOUVELLE INDEXATION
C      INDEX=ITAB(4)
C      MISE EN TABLEAU DU TITRE
C*****
DO 12 I=1,10
12 LIB(I)=LTT(I)
IDJ=ITAB(2)/4096
IDF=ITAB(2)-IDJ*4096
IUJ=IDF/256
IDF=IDF-IUJ*256
IDATE(1)=IDJ*10+IUJ
IDM=IDF/16
IUM=IDF-IDM*16
IDATE(2)=IDM*10+IUM
IDATE(3)=79
DO 20 I=1,6
20 IDB1(INDEX,I)=ITAB(I)
DO 22 I=1,4
22 DB1(INDEX,I)=TAB(I)
DO 24 I=1,ITAB(5)
24 TETA(INDEX,I)=ANG(I)
C      REMISE EN ORDRE DES SPECTRESB SI BANDE EN SENS INVERSE

```

```

      DO 250 LA=1,ITAB(5)
      IF(ITAB(6).GT.1) IA=ITAB(5)+1-LA;GO TO 170
      IA=LA
C      LECTURE DES SPECTRES
170 CONTINUE
      READ(ND,105,END=630)(TAB2(I),I=1,50)
C      RANGEMENT DES ANGLES ET DES SPECTRES
      OMEGA(INDEX,IA)=ANGLE
      DO 250 IDB=1,45
250 SDB(INDEX,IA,IDB)=SPE(IDB)
      NUMER=NUMER+1
      IF(NUMER.LE.8) GO TO 50
      GO TO 700
500 DO 25 J=1,ITAB(5)
      READ(ND,105,END=620)(TAB2(I),I=1,50)
      25 CONTINUE
      GO TO 600
610 KODR=1
      GO TO 650
620 KODR=2
      GO TO 650
630 KODR=3
650 WRITE(6,104)KODR;STOP
104 FORMAT(' ERREUR EN LECTURE BANDE KODR=',I5)
700 CONTINUE
C      MISE EN TABLEAU DES FREQ DES ANGLES ET SPECTRES DE 0A14 METRES
C      COEFFICIENTS DE CALIBRATION LIGNE 1:C11 C12 ,LIGNE 2:C21 C22
C      RANG DU PREMIER ANGLE LIGNE 1 ET LIGNE 2
      READ(IUN,901)C11,C12,C21,C22,IRG(1),IRG(2)
901 FORMAT(4F5.1,2I2)
      WRITE(6,801)C11,C12,C21,C22
801 FORMAT('/ CALIBRATION L1 L2 =',4F6.1)
C      RANG-1 DE LA PREMIERE FREQ TRAITEE
      NFDEC=14
      NFREQ=25
      DO 5 IDB=1,NFREQ
5      FREQ(IDB)= JFR(IDB+NFDEC)
C      CHOIX DU TRAITEMENT
C      KMIC=0 MICROS A+ B MOYENNES
C      KMIC=1 MICRO A
C      KMIC=2 MICRO B
C
C
      IA=IB=0
C      BOUCLE SUR LES LIGNES
      DO 800 KLIQ=1,2
      IF(KMIC.NE.0.AND.KMIC.NE.1)GO TO 1000
C      MICRO A
      IF(KLIQ.EQ.1)M1=IPI(1);M2=IPI(5);NT1=IDB1(M1,5)
      IF(KLIQ.EQ.2)M1=IPI(3);M2=IPI(7);NT2=IDB1(M1,5)
      I=1
      LEGAL=0
      DO 10 K=1,IDB1(M1,5)
      K1=INT(OMEGA(M1,K)+0.5)
      K2=INT(OMEGA(M2,I)+0.5)
      IF(K1.EQ.K2) I=I+1;LEGAL=LEGAL+1
10 CONTINUE
      DO 30 I=IRG(KLIQ),IDB1(M1,5)
      IA=IA+1
      ANGLES(IA)=OMEGA(M1,I)

```

```

      DO 30 IDB=1,NFREQ
30  DECB (IA,IDB)=SDB(M1,I ,IDB+NFDEC)
      DO 40 J=LEGAL +1,IDB1(M2,5)
      IA=IA+1
      ANGLES(IA)=OMEGA(M2,J)
      DO 40 IDB=1,NFREQ
40  DECB (IA,IDB)=SDB(M2,J,IDB+NFDEC)
      IF(KMIC.NE.0)GO TO 2000
C    MICRO B
1000 IF(KLIG.EQ.1)M1=IPI(2);M2=IPI(6);NT1=IDB1(M1,5)
      IF(KLIG.EQ.2)M1=IPI(4);M2=IPI(8);NT2=IDB1(M1,5)
      I=1
      LEGAL=0
      DO 11 K=1,IDB1(M1,5)
      K1=INT(OMEGA(M1,K)+0.5)
      K2=INT(OMEGA(M2,I)+0.5)
      IF(K1.EQ.K2) I=I+1;LEGAL=LEGAL+1
11  CONTINUE
      DO 31 I=IRG(KLIG),IDB1(M1,5)
      IB=IB+1
      ANGLES(IB)=OMEGA(M1,I)
      DO 31 IDB=1,NFREQ
31  AUX(IB,IDB)=SDB(M1,I,IDB+NFDEC)
      DO 41 J=LEGAL+1,IDB1(M2,5)
      IB=IB+1
      ANGLES(IB)=OMEGA(M2,J)
      DO 41 IDB=1,NFREQ
41  AUX(IB,IDB)=SDB(M2,J,IDB+NFDEC)
      IF(KMIC.NE.0)GO TO 3000
      IF(KLIG.EQ.1)MIKEA=IA;GO TO 800
      MIKES=IA
      MIKEB=MIKES-MIKEA
C    MOYENNE MICRO A + MICRO B
      DO 70 I=1,MIKES
      DO 70 IDB=1,NFREQ
70  DECB(I,IDB)=(DECB(I,IDB)+AUX(I ,IDB))/2.
      GO TO 5000
2000 IF(KLIG.EQ.1)MIKEA=IA;GO TO 800
      MIKES=IA
      MIKEB=MIKES-MIKEA
      GO TO 5000
3000 IF(KLIG.EQ.1)MIKEA=IB;GO TO 800
      MIKES=IB
      MIKEB=MIKES-MIKEA
      DO 72 I=1,MIKES
      DO 72 IDB=1,NFREQ
72  DECB(I,IDB)=AUX(I,IDB)
800  CONTINUE
5000 IF(MIKES.GT.130)WRITE(6,106)MIKES;STOP
106  FORMAT(' TROP D ANGLES LIGNE 1 + LIGNE 2 =',I5)
C    CORRECTION DES SPECTRES
      DO 73 I=1,NT1
      DO 73 IDB=1,NFREQ
73  DECB(I,IDB)=DECB(I,IDB)+C11
      DO 74 I=NT1+1,MIKEA
      DO 74 IDB=1,NFREQ
74  DECB(I,IDB)=DECB(I,IDB)+C12
      DO 75 I=MIKEA+1,MIKEA+NT2
      DO 75 IDB=1,NFREQ
75  DECB(I,IDB)=DECB(I,IDB)+C21

```

```

      DO 76 I=MIKEA+NT2+1,MIKES
      DO 76 IDB=1,NFREQ
76  DECBI(1,IDB)=DECB(1,IDB)+C22
      RETURN
      END
*DECK CORECT
      SUBROUTINE CORECT
C
C      ICALL = 1  FOR MIKE RESPONSE CORRECTIONS
C      ICALL = 2  FOR REVERBERATION CORRECTIONS
C      ICALL = 3  FOR BACKGROUND NOISE CORRECTIONS
C
C
C*****
      REAL*8  DECB,ANGLES,XDIS,HTMIKE,HTSRCE,SDHUM,SDTEMP,V8,UNITS,
1          THETAM,ADEL,BDEL,AN,YF,YY,ANHI,ANLO,AFHI,AFL0,
2          COFA,COFB,TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,VJET,
3          FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C*****
      REAL*8  CORVAL,DUMMY1,DUMMY2
C*****
      COMMON /SUB/ MIKES,MIKEA,MIKEB,ICALL,DECB(130,35),
1          ANGLES(130),XDIS(130,2),HTMIKE(130),HTSRCE,SDHUM,
2          SDTEMP,V8,UNITS,KFIT(2),IWTB,IWTE,
3          THETAM(33),ADEL,BDEL,AN,IGRC,NRUN,
4          YF,YY,ANHI,ANLO,AFHI,AFL0,COFA(16,35),COFB(16,35),
5          TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,
6          IBID0,IMRC,IBNC,IRC,IAAC,IDATE,
7          VJET,FREQ(33),THETAS(50,33),XOVERD(50,33),NTAB(33),
8          NTEST,XDMAX(33),THETSM(33),RNZDIA,NZTYPE(10),NFREQ
C
C      REAL*8  CORR(130,33)
C
C
C      INITIALIZE ARRAY OF CORRECTION VALUES
C
      DO 100 J=1,33
      DO 100 K=1,50
100  CORR(K,J) = 0.0
C
      REWIND 8
      GO TO (200,400,600),ICALL
C
C
C      THIS SECTION IS USED FOR MIKE RESPONSE CORRECTIONS
C
200  READ (8) IRUN,ICHAN,IBAND,CORVAL,DUMMY1,DUMMY2
      IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GO TO 300
      IF (ICHAN.LT.1.OR.ICHAN.GT.50) GO TO 200
      IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GO TO 200
      CORR(ICHAN,IBAND) = CORVAL
      IF ((ICHAN+MIKEA).GT.50) GO TO 200
      IM = ICHAN+MIKEA
      CORR(IM,IBAND) = CORVAL
      GO TO 200
C
300  WRITE(6,801)
801  FORMAT (1H1,10X,31HMIKE RESPONSE CORRECTION VALUES /)
      CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)

```

```

C
C   APPLY CORRECTIONS
C
250 DO 225 J=1,NFREQ
    DO 225 K=1,MIKES
225  DECB(K,J) = DECB(K,J)-CORR(K,J)
    GOTO 900
C
C   THIS SECTION APPLIES REVERBERATION CORRECTIONS
C
C
400 READ (8) IRUN,ICHAN,IBAND,DUMMY1,CORVAL,DUMMY2
    IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GOTO 500
    IF (ICHAN.LT.1.OR.ICHAN.GT.50) GOTO 400
    IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GOTO 400
    CORR(ICHAN,IBAND) = CORVAL
    IF ((ICHAN+MIKEA).GT.50) GOTO 400
    IM = ICHAN+MIKEA
    CORR(IM,IBAND) = CORVAL
    GOTO 400
C
500 WRITE(6,802)
802  FORMAT (1H1,10X,31HREVERBERATION CORRECTION VALUES /)
    CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
    GOTO 250
C
C   APPLY BACKGROUND NOISE CORRECTIONS HERE
C
C
600 READ (8) IRUN,ICHAN,IBAND,DUMMY1,DUMMY2,CORVAL
    IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GOTO 700
    IF (ICHAN.LT.1.OR.ICHAN.GT.50) GOTO 600
    IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GOTO 600
    CORR(ICHAN,IBAND) = CORVAL
    IF ((ICHAN+MIKEA).GT.50) GOTO 600
    IM = ICHAN+MIKEA
    CORR(IM,IBAND) = CORVAL
    GOTO 600
C
700 WRITE (6,803)
803  FORMAT (1H1,10X,34HBACKGROUND NOISE CORRECTION VALUES /)
    CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
    GOTO 250
C
C   SUBROUTINE EXIT
C
900 RETURN
    END
*DECK AIFAB
    SUBROUTINE AIFAB (P,T,RH,CF,ABDBM)
C
C   -----
C   THIS PROGRAM CALCULATES THE ABSORPTION OF SOUND IN AIR AS A
C   FUNCTION OF TEMPERATURE, HUMIDITY, PRESSURE AND FREQUENCY.
C   THE PROGRAM SHOULD NOT BE USED FOR CALCULATIONS OUTSIDE
C   THE TEMPERATURE RANGE OF 0 DEG F (-20 DEG C) THROUGH 104 DEG F
C   (40 DEG C)
C

```

```

C      ABDBM = AMPLITUDE ABSORPTION COEFFICIENT IN DB/METER
C      ABDBSC = AMPLITUDE ABSORPTION COEFFICIENT IN DB/SECONDS
C      ABDBTF = AMPLITUDE ABSORPTION COEFFICIENT IN DB/1000FT
C      ABDAM = AMPLITUDE ABSORPTION COEFFICIENT PER WAVELENGTH
C      ALPHA = AMPLITUDE ABSORPTION COEFFICIENT IN NEPERS PER METER
C
C      CF = ACOUSTIC FREQUENCY
C      P = AMBIENT PRESSURE IN ATM
C      PS = SATURATED VAPOR PRESSURE FOR WATER IN ATMOSPHERES
C      T = TEMPERATURE IN DEGREES KELVIN
C      TC = TEMPERATURE IN DEGREES CENTIGRADE
C      TF = TEMPERATURE IN DEGREES FAHRENHEIT
C      WAVEL = WAVELENGTH OF SOUND WAVE
C      -----
C
C*****
      REAL*8 P,T,RH,CF,ABDBM
      REAL*8 P1,T1,TC,TF,VEL,VELFPS,T01,PS,H,FR02,FRN2,
1          ALPHA,WAVEL,ABDAM,ABDBTF,ABDBSC
C*****
      P1 = 3.14159
      T1 = T/293.
      TC = T-273.
      TF = TC*1.8+32.
      VEL = 343.4*DSQRT(T1)
      VELFPS = VEL*3.28
      T01 = 273.16
C
      PS = 10.79586*(1.-T01/T)-5.02808*DL0G10(T/T01)+1.50474E-4*(1.-10.
C**(-8.29692*((T/T01)-1.)))+0.42873E-3*(10.**((4.76955*
C(1.-(T01/T)))-1.))-2.2195983
      PS = 10.**PS
      H = PS/P*RH
      FR02 = P*(24.+4.41E04*H*(0.05+H)/(0.391+H))
      FRN2 = P/DSQRT(T1)*(9.+350.*H*DEXP(-6.142*((1./T1)**.331-1.)))
      ALPHA = 1.84E-11+2.1913E-4/T1*P*(2239.1/T)**2*DEXP(-2239.1/T)
C/(FR02+(CF**2/FR02))
      ALPHA = ALPHA+8.1619E-4/T1*P*(3352./T)**2*DEXP(-3352./T)
C/(FRN2+(CF**2/FRN2))
      ALPHA = ALPHA*DSQRT(T1)*CF**2/P
      WAVEL = VEL/CF
      ABDAM = ALPHA*WAVEL
      ABDBTF = ALPHA*2647.
      ABDBM = ALPHA*8.6860
      ABDBSC = ALPHA*VEL*8.686
      CONTINUE
      RETURN
      END
*DECK GRN01S
      SUBROUTINE GRN01S (XDIST,HDIST,JETN,H,TF,MIKES,FI,NFREQ)
C
C      CORRECTIONS FOR EACH MIKE DISTANCE AND FREQUENCY
C      SUBROUTINE TO COMPUTE GROUND REFLECTIONS
C
C*****
      REAL*8 P12,DFFI,TT,A,B,C,X,HMIKE,RPRIME,RSMALL,Z,DR,R
C*****
      REAL*8 L1
      REAL*8 JETN(130,35),XDIST(130),HDIST(130),FI(33),L(24)
      REAL*8 TF,H

```

```

C*****
C      DATA P12,DFF1/6.2831853071796,0.115/
C
C      C = SPEED OF SOUND
C      DR = DIFFERENCE BETWEEN DIRECT PATH AND REFLECTED PATH FOR SOUND
C      H = SOURCE HEIGHT ABOVE GROUND
C      HMIKE = MIKE HEIGHT ABOVE GROUND (HDIST)
C      JETN IS ARRAY CONTAINING REFLECTION INDEX
C      L1 = CENTER FREQUENCY WAVE LENGTH
C      R IS THE RATIO OF AVERAGE QUADRATIC VALUE OF THE RESULTING
C      SIGNAL AND THAT OF THE MAIN SIGNAL.
C      TF = TEMPERATURE, DEGREES FARENHEIT
C      TT = TEMPERATURE, DEGREES RANKINE
C      X = GROUND DISTANCE BETWEEN SOURCE AND MICROPHONE
C      Z = RATIO OF REFLECTED PATH TO DIRECT PATH OF SOUND WAVE
C      WHEN F1(1) = CENTER FREQUENCY OF 1/3 OCTAVE BAND THEN THE BAND
C      WIDTH IS .23 TIMES THE CENTER FREQUENCY. FOR INSTANCE THE BAND
C      WIDTH FOR 50 HZ WOULD BE 11.5. OLD EQUATION WAS DF = .23 * F1(1)
C      DFF1 = DF/(1. * F1(1) WHICH SIMPLIFIES TO DFF1 = 0.115
C
C      TT = TF + 459.7
C      A = P12 * DFF1
C      B = P12 * DSQRT(1.0 + DFF1 * DFF1)
C      C = 49.01* DSQRT(TT)
C      DO 10 I=1,NFREQ
10  L(I) = C/F1(I)
C
C      DO 100 M = 1,MIKES
C      X = XDIST(M)
C      HMIKE = HDIST(M)
C      RPRIME = DSQRT((H + HMIKE)**2 + X**2)
C      RSMALL = DSQRT((H - HMIKE)**2 + X**2)
C      Z = RPRIME/RSMALL
C      DR = RPRIME - RSMALL
C      IF (DR.EQ.0.0) DR=0.001
C      IF (Z.EQ.0.0) Z = 0.001
C      DO 50 I=1,NFREQ
C      L1 = L(I)
C      R = 1.+(1/Z**2) + 2*DSIN(A*(DR/L1)) *DCOS(B*(DR/L1))/(Z*(A*(DR/L1)
C      *))
C      IF (R.LT.0.0) GOT0 50
C      JETN(M,1) = 10.0*DLOG10(R)
50  CONTINUE
100 CONTINUE
C
C      RETURN
C      END
C      *DECK OUTCOR
C      SUBROUTINE OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
C
C      THIS SUBROUTINE PRINTS OUT THE ARRAY 'CORR' FOR NEAR AND FAR FIELDS .
C
C*****
C      REAL*8 CORR(130,33)
C
C      400 FORMAT(1H1)
C
C      MIC = MIKEA
C

```

```

C
C      J=1 FOR NEAR-SIDELINE MIKES (MIKEA).
C      J=2 FOR FAR-SIDELINE MIKES (MIKEB).
C
C      IF ( MIC .GT. 13 ) MIC = 13
C      MICADD = MIC
C      MICNUM = MIC
C      M1 = 1
C
C      DO 2000 J = 1 , 2
C
C      IF ( J .EQ. 1 ) WRITE ( 6,430)
430  FORMAT (1H0,50X,23HNEAR FIELD MICROPHONES. /)
C      IF ( J .EQ. 2 ) WRITE ( 6,460)
460  FORMAT ( 1H1,50X,23H FAR FIELD MICROPHONES. / )
C
C      N1 = 1
C      MICNUM = MICADD
C
C      L=1 FOR THE 1ST 13 MIKES.
C      L=2 FOR THE REST OF THE MIKES.
C
C      DO 1500 L = 1 , 2
C
C      IF(L.EQ.1)WRITE(6,500) (I,I = N1 , MICNUM )
500  FORMAT(5X,8HMIKE , 8(11,8X),11,04(7X,12) )
C      IF(L.EQ.2)WRITE(6, 550) (I,I = N1 , MICNUM )
550  FORMAT ( 5X,7HMIKE , 12,11(7X,12) )
C      WRITE(6,600)
600  FORMAT(1X,4HBAND , / )
C
C
C      'K' REFERS TO THE MIKE NUMBER.
C      'I' REFERS TO THE BAND NUMBER.
C
C      DO 1000 I = 1 , NFREQ
C
C      WRITE(6,900) I , (CORR(K,I) , K = M1,MIC )
900  FORMAT(2X,12,4X,F7.2,12(2X,F7.2) )
1000 CONTINUE
C
C      IF ( MIKEA .EQ. MICADD ) GO TO 1700
C      M1 = MIC + 1
C      N1 = MICNUM + 1
C      MIC = MIKEA * J
C      MICNUM = MIKEA
C
C
C      WRITE(6,400)
C
C      1500 CONTINUE
C
C      1700 CONTINUE
C
C      N1 = 1
C      M1 = MIKEA + 1
C      MIC = MIKEA + MICADD
C
C      2000 CONTINUE

```



```

        WRITE(6,400)
C
      RETURN
      END
*DECK POLYX
      REAL*8 FUNCTION POLYX(COF,KFIT,X)
C
      EVALUATE POLYNOMIAL OF ARBITRARY ORDER
      USING HORNER'S RULE.
C
      COF - COEFFICIENTS OF POLYNOMIAL
      POLYX = COF(1) + COF(2)*X + COF(3)*X**2 + .....
      (KFIT + 1 = NUMBER OF COEFFICIENTS.)
C
      KFIT - HIGHEST POWER OF POLYNOMIAL
C
      X - VALUE OF INDEPENDENT VARIABLE AT WHICH
      POLYNOMIAL IS TO BE EVALUATED.C
C*****
      REAL*8 COF(16)
      REAL*8 SUM,X
C*****
C
      NP1=KFIT + 1
      SUM = COF(NP1)
      IF (KFIT.EQ.0) GOTO 20
C
      DO 10 I=1,KFIT
      J=NP1-I
      SUM=COF(J)+SUM*X
      10 CONTINUE
C
      20 POLYX = SUM
      RETURN
      END
*DECK COMP
      SUBROUTINE COMP (Y1,Y2,TI1,TI2,TS,X1)
C
C*****
      REAL*8 D2R,TERM1,TERM2,TERM3,P2,P1,ADD
      REAL*8 TI1,TI2,TS,X1,Y1,Y2
C*****
      DATA D2R /0.017453293/
C
C
      TERM1 = DTAN((180.-TI2)*D2R)
      TERM2 = DTAN((180.-TI1)*D2R)
      IF (TERM1.EQ.0..OR.TERM2.EQ.0.) GOTO 999
      P2 = (Y2/TERM1)-(Y1/TERM2)
      TERM3 = (Y2/Y1)-1.0
      IF (TERM3.EQ.0.) GOTO 999
      P1 = P2/TERM3
      X1 = (Y1/TERM2)-P1
      IF (P1.LT.0.) ADD=0.
      IF (P1.GE.0.) ADD=180.
      TS = -1.0*(DATAN(Y1/P1)/D2R) + ADD
      RETURN
C
      999 X1 = 0.0
      TS = 0.0

```

```

      RETURN
      END
*DECK DIRECT
      SUBROUTINE DIRECT
C
C
C      DETERMINE THE NEAR FIELD/FAR FIELD
C      DIRECTIVITY RELATIONSHIP
C
C*****
      REAL*8 DECB, ANGLES, XDIS, HTMIKE, HTSRCE, SDHUM, SDTEMP, V8, UNITS,
1      THETAM, ADEL, BDEL, AN, YF, YY, ANHI, ANLO, AFHI, AFLQ,
2      COFA, COFB, TR, R2D, D2R, SDTEMP, TTEMP, TTEMPF, VJET,
3      FREQ, THETAS, XOVERD, XDMAX, THETSM, RNZDIA
C*****
      REAL*8 STR, DFMAX, XPEAKN, XPEAKF
      REAL*8 XMAXN, YMAXN, XMAXF, YMAXF, TS, X1, DIFMAX, S, XNEAR,
1      XFAR, FACT, XGUESS, YNEAR, YFAR
C*****
      COMMON /SUB/ MIKES, MIKEA, MIKEB, ICALL, DECB(130, 35),
1      ANGLES(130), XDIS(130, 2), HTMIKE(130), HTSRCE, SDHUM,
2      SDTEMP, V8, UNITS, KFIT(2), IWTB, IWTE,
3      THETAM(33), ADEL, BDEL, AN, IGRC, NRUN,
4      YF, YY, ANHI, ANLO, AFHI, AFLQ, COFA(16, 35), COFB(16, 35),
5      TR, R2D, D2R, SDTEMP, TTEMP, TTEMPF,
6      IBIDQ, IMRC, IBNC, IRC, IAAC, IDATE,
7      VJET, FREQ(33), THETAS(50, 33), XOVERD(50, 33), NTAB(33),
8      NTEST, XDMAX(33), THETSM(33), RNZDIA, NZTYPE(10), NFREQ
      COMMON/TABD/STR(33), DFMAX(33), XPEAKN(33), XPEAKF(33)
C
      REAL*8 XTF(50, 2), XTN(50, 2)
      DIMENSION NANGT(2)
C
C      NFREQ = NUMBER OF FREQUENCIES UNDER CONSIDERATION
C
      NANGT(1) = 0
      NANGT(2) = 0
      DO 2000 N=1, NFREQ
      M = 0
C
C      FIND PEAK SPL'S FOR A GIVEN FREQUENCY
C      FOR BOTH NEAR FIELD AND FAR FIELD.P
C      (CONSIDER GASPL AND PNL JUST ANOTHER SET OF
C      FREQUENCY DATA)
C
      CALL FNDMAX(COFA(1, N), KFIT(2), ANHI, ANLO, XMAXN, YMAXN, NQMAX)
      IF (NQMAX.EQ.1) GOTO 1800
      CALL FNDMAX(COFB(1, N), KFIT(2), AFHI, AFLQ, XMAXF, YMAXF, NQMAX)
      IF (NQMAX.EQ.1) GOTO 1800
C
C      COMPUTE X/D AND THETAS MAX VALUES FOR
C      PEAKS OF CURVES OF N-TH FREQUENCY
C
      CALL COMP(AN, YF, XMAXN, XMAXF, TS, X1)
      XDMAX(N) = X1/RNZDIA
      THETSM(N) = TS
      DIFMAX=YMAXN-YMAXF
      S = FREQ(N)*RNZDIA/V8
      DFMAX(N)=DIFMAX
      STR(N)=S

```

```

XPEAKN(N)=XMAXN
XPEAKF(N)=XMAXF
WRITE (6,1002) S,FREQ(N),DIFMAX,AN,ADEL,YF,BDEL,XMAXN,XMAXF,
1 THETSM(N),XDMAX(N)
C
C      GO THROUGH THIS PROCESS TWICE.
C      L = 1 TO RIGHT OF PEAK
C      L = 2 TO LEFT OF PEAK.
C
DO 1000 L=1,2
XNEAR=XMAXN
XFAR=XMAXF
NANG=0
FACT=1.0
IF(L.EQ.2)FACT=-1.0
200 CONTINUE
C
C      TAKE A STEP IN THE NEAR FIELD
C      AWAY FROM THE PEAK
C
XNEAR=XNEAR + FACT*ADEL
IF(XNEAR.GT.ANHI.OR.XNEAR.LT.ANLO)GO TO 500
XGUESS=XFAR + FACT*BDEL
C
C      FIND THE CORRESPONDING NEAR FIELD SPL
C
YNEAR=POLYX(COFA(1,N),KFIT(2),XNEAR)
C
C      FIND THE FAR FIELD ANGLE WHICH GIVES
C      THE PROPER NEAR FIELD/FAR FIELD SPL DIFFERENCE
C
YFAR=YNEAR-DIFMAX
CALL FINDY(COFA(1,N),KFIT(2),XFAR,YFAR,XGUESS,XMAXF,
1 AFL0,AFHI,FACT,K0)
IF(K0.EQ.0)GO TO 500
C
C      IF K0 .NE. 0 WE FOUND THE PROPER ANGLE.
C
NANG=NANG+1
C
C      SAVE ANGLES IN TEMPORARY ARRAYS BECAUSE
C      THEY ARE IN THE WRONG ORDER FOR PLOTTING,
C      REARRANGE THEM LATER.
C
XTF(NANG,L)=XFAR
XTN(NANG,L)=XNEAR
NANGT(L)=NANG
IF (NANG.EQ.50) GOTO 1000
GO TO 200
500 CONTINUE
C
C      K0 = 0 THEREFORE WE EITHER RAN OUT OF ANGLES OR COULDN'T FIND ONE
C      OR NEAR FIELD ANGLE REQUESTED WAS OUT OF RANGE.
C
1000 CONTINUE
C
C      REARRANGE ANGLE ARRAYS TO PUT IN PROPER ORDER
C
N1=NANGT(2)
IF(N1.EQ.0)GO TO 1600

```

```

C
C      DO LEFT OF PEAK FIRST
C
      DO 1500 J=1,N1
      K=N1+1-J
      M=M+1
C      COMPUTE X/D AND THETAS AND STORE
      CALL COMP(AN,YF,XTN(K,2),XTF(K,2),TS,X1)
      XOVERD(M,N) = X1/RNZDIA
      THETAS(M,N) = TS
      WRITE(6,1004) M,XTN(K,2),XTF(K,2),TS,XOVERD(M,N)
      IF (M.EQ.50) GOTO 1800
1500 CONTINUE
1600 CONTINUE
C
C      DO RIGHT OF PEAK NEXT
      N2=NANGT(1)
      IF(N2.EQ.0)GOTO 1800
      DO 1700 J=1,N2
      M=M+1
C      COMPUTE X/D AND THETAS AND STORE
      CALL COMP(AN,YF,XTN(J,1),XTF(J,1),TS,X1)
      XOVERD(M,N) = X1/RNZDIA
      THETAS(M,N) = TS
      WRITE(6,1004) M,XTN(J,1),XTF(J,1),TS,XOVERD(M,N)
      IF (M.EQ.50) GOTO 1800
1700 CONTINUE
1800 CONTINUE
      NTAB(N)=M
      IF (M.EQ.0) WRITE(6,1001) FREQ(N)
      IF (M.NE.0) WRITE(6,1003) M
2000 CONTINUE
C
      RETURN
C
1001 FORMAT(1H0,27HNO PEAK FOUND FOR FREQUENCY,F9.0)
1002 FORMAT(1H1 / 10X,14HSTROUHAL NO. = , F7.2 /10X,11HFREQUENCY = ,
1 F10.1//10X,17HDIFFERENCE MAX = ,F6.2// 10X,
2 15HNEAR SIDELINE = ,F6.1,3H FT,16X,6HDEL = ,F6.2 / 10X,
3 15HFAR SIDELINE = ,F6.1,3H FT,16X,6HDEL = ,F6.2 // 26X,4HNEAR,
4 10X,3HFAR / 10X,10HPEAK (DEG),F11.2,2X,F11.2 / / 10X,7HTHETA-S,
5 3X,F11.2, / 10X,3HX/D,7X,F11.2, // 1X,
6 33HRESULTS OF TABULATED ANGLE PAIRS //20X,4HNEAR,8X,3HFAR /
7 10X,3HNO. ,1X,2(6X,5HTHETA),6X,7HTHETA-S,8X,3HX/D / 9X,
8 4(1H-),2(5X,7(1H-)),4X,9(1H-),6X,5(1H-) /)
1003 FORMAT( // 5X,6HNTAB = ,I3)
1004 FORMAT(10X,12,6X,F7.2,5X,F7.2,5X,F6.2,7X,F6.2)
C
      END
      SUBROUTINE STAB(NFREQ,FREQ,THETSM,XDMAX)
C*****
      REAL*8 STR,DFMAX,XPEAKN,XPEAKF
C*****
      COMMON/TABD/STR(33),DFMAX(33),XPEAKN(33),XPEAKF(33)
      REAL*8 FREQ(1),THETSM(1),XDMAX(1)
C      ECRITURE RESULTATS
      WRITE(6,105)
      DO 10 I=1,NFREQ
10 WRITE(6,101)STR(I),FREQ(I),THETSM(I),XDMAX(I),DFMAX(I),

```

```

      1XPEAKN(1),XPEAKF(1)
105  FORMAT(1H1,' STROUHAL FREQUENCE  THETAS      X/D      DIFMAX  PEAKN
      1  PEAKF (DEG)')
101  FORMAT(1X,7F9.2)
      RETURN
      END
*DECK FINDY
      SUBROUTINE FINDY(COF,KFIT,XFIND,YFIND,XGUESS,XMAX,XLO,XHI,
      - DRCTN,KO)
C
C
C -----
C
C      GIVEN A POLYNOMIAL OF ORDER KFIT IN 1 INDEPENDENT VARIABLE, THIS
C      SUBROUTINE WILL DETERMINE THE VALUE OF THE INDEPENDENT VARIABLE (X)
C      CORRESPONDING TO A KNOWN VALUE OF THE DEPENDENT VARIABLE (Y).  X IS
C      FOUND VIA ITERATION USING THE NEWTON-RAPHSON METHOD OF FINDING
C      ROOTS OF EQUATIONS IN ONE VARIABLE.
C
C      COF = ARRAY OF COEFFICIENTS OF X IN INCREASING ORDER
C      KFIT = HIGHEST DEGREE (ORDER) OF X
C      XFIND = THE VALUE FOUND FOR X
C      YFIND = THE VALUE OF Y FOR WHICH X IS TO BE DETERMINED
C      XGUESS = INITIAL GUESS FOR X (MAY OR MAY NOT CONVERGE)
C      XLO = LOWER LIMIT VALUE FOR X
C      XHI = HIGHER LIMIT VALUE FOR X
C      XMAX = LARGEST VALUE OF X OVER THE INTERVAL XLO,XHI
C      DRCTN = POSITIVE IF SLOPE OF CURVE IS POS & NEG IF SLOPE IS NEG
C      KO = 1  IF THE VALUE OF X IS SUCCESSFULLY FOUND
C           0  IF THE ITERATION DIVERGES & NO X VALUE IS FOUND
C
C -----
C
C *****
C      REAL*8 XFIND,YFIND,XGUESS,XMAX,XLO,XHI,DRCTN
C      REAL*8 XN,DELTA,C1,FX,FXD,XN1,DEL,XB,XE,FXD2,AGUESS
C *****
C      COMMON /OUTFLO/ IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
C
C      REAL*8 COF(16),COFD(16),COFD2(16)
C
C      INITIALIZE PROGRAM VARIABLES
C      KO = 1
C      IFLAG = 0
C      XN = XGUESS
C      I1 = KFIT+1
C      KFID = KFIT-1
C      DELTA = 1.0E-6
C      DO 50 K=1,2
C      COFD(K) = 0.0
C 50  COFD2(K) = 0.0
C
C      FIND THE FIRST DERIVATIVE OF F(X) AND PLACE THE CALCULATED
C      COEFFICEIENTS INTO COFD ARRAY IN ASCENDING ORDER OF POWERS
C      OF THE INDEPENDENT VARIABLE.
C
C      DO 100 I=2,I1
C      COFD(I-1) = COF(I)*(I-1)
100  CONTINUE
C      C1 = COF(1)

```



```

500 CCF(1) = C1
    IF (IPFNDY.EQ.0) GOTO 800
    WRITE(6,700) XFIND,YFIND,K0
700  FORMAT(1X,9H$UB FINDY,6X,7H$FIND =,F11.6,4X,7H$FIND =,F11.6,4X,
-      4H$0 =,14)
800  RETURN
    END
*DECK FNDMAX
SUBROUTINE FNDMAX(CCF,KFIT,XHI,XLO,XMAX,YMAX,NOMAX)
C
C -----
C   GIVEN A POLYNOMIAL OF ARBITRARY ORDER, THIS SUBROUTINE WILL FIND THE
C   VALUE OF THE INDEPENDENT VARIABLE (X) WHICH, WHEN EVALUATED, GIVES
C   THE MAXIMUM POSITIVE VALUE OF THE DEPENDENT VARIABLE (Y) OVER THE
C   SPECIFIED INTERVAL (XLO <= X <= XHI).
C
C
C   CCF = POLYNOMIAL COEFFICIENTS IN ASCENDING ORDER
C   KFIT = DEGREE OF FIT OF POLYNOMIAL (ORDER)
C   XHI = HIGHEST VALUE WHICH X CAN TAKE ON
C   XLO = LOWEST VALUE WHICH X CAN TAKE ON
C   XMAX = MAXIMUM VALUE OF X OVER THE INTERVAL SPECIFIED BY XLO,XHI
C   YMAX = HIGHEST VALUE OF Y OVER THE INTERVAL FOR X
C   NOMAX = FLAG - 1 = NO MAX VALUE FOUND FOR X, 0 = MAX VALUE FOUND OK
C   D = COEFFICIENTS OF THE 1ST DERIVATIVE (ASCENDING ORDER)
C   ROOT = ROOTS OF THE 1ST DERIVATIVE
C -----
C*****
REAL*8 XHI,XLO,XMAX,YMAX
REAL*8 R,Y
C*****
COMMON /OUTFLO/ IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
REAL*8 CCF(16),D(16),ROOT(16)
NOMAX = 0
J1 = KFIT+1
C
C
C   GENERATE 1ST DERIV FOR THE GIVEN POLYNOMIAL
C
L = 0
DO 200 J = 2,J1
L = L + 1
D(L) = L * CCF(J)
200 CONTINUE
C
C   FIND ALL THE ROOTS OF THE DERIVATIVE
C
N = KFIT-1
CALL ZEROS(N,D,XLO,XHI,ROOT,NROOTS,IER,IPROOT)
IF (IER.NE.0) GOTO 700
C
C   FOR EACH ROOT IN THE INTERVAL , TEST FOR MAXIMUM Y VALUE
C
XMAX = -1E5
YMAX = -1E5
DO 300 I=1,N
C
R = ROOT(I)
C

```

```

      IF (R.GT.XHI.OR.R.LT.XLO) GOTO 300
      Y = POLYX(COF,KFIT,R)
      IF (Y.LT.YMAX) GOTO 300
      XMAX = R
      YMAX = Y
300  CONTINUE
C
C      TEST THE ENDP0INTS OF THE INTERVAL.  IF NO ROOTS OF THE
C      DERIVATIVE FALL WITHIN THE INTERVAL, THEN ONE OF THE ENDP0INTS
C      CAN BE A LOCAL MAXIMUM, OR IF THE CURVE IS INCREASING ON THE
C      INTERVAL AFTER A LOCAL MAX, THE ENDP0INT MAY BE A MAX FOR THE INTERVAL.
C
      Y = POLYX(COF,KFIT,XLO)
      IF (Y.LT.YMAX) GOTO 400
      XMAX = XLO
      YMAX = Y
400  Y = POLYX(COF,KFIT,XHI)
      IF (Y.LT.YMAX) GOTO 500
      XMAX = XHI
      YMAX = Y
700  NOMAX = 1
500  CONTINUE
      IF (IPFNDM.EQ.0) GOTO 600
      WRITE(6,501) XMAX,YMAX,NOMAX
501  FORMAT(1X,10H$UB FNDMAX,5X,6HXMAX =,F10.5,4X,6HYMAX =,F10.5,4X,
-       7HNOMAX =,14)
600  RETURN
      END
*DECK KURVFT
      SUBROUTINE KURVFT(KLISF)
C
C*****
      REAL*8 DECB,ANGLES,XDIS,HTMIKE,HTSRCE,SDHUM,SDTEMP,V8,UNITS,
1      THETAM,ADEL,BDEL,AN,YF,YY,ANH1,ANL0,AFH1,AFLO,
2      COFA,COFB,TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,VJET,
3      FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C*****
      COMMON /SUB/ MIKES,MIKEA,MIKEB,ICALL,DECB(130,35),
1      ANGLES(130),XDIS(130,2),HTMIKE(130),HTSRCE,SDHUM,
2      SDTEMP,V8,UNITS,KFIT(2),IWTB,IWTE,
3      THETAM(33),ADEL,BDEL,AN,IGRC,NRUN,
4      YF,YY,ANH1,ANL0,AFH1,AFLO,COFA(16,35),COFB(16,35),
5      TR,R2D,D2R,SDTEMF,TTEMP,TTEMPF,
6      IBID0,IMRC,IBNC,IRC,IAAC,IDATE,
7      VJET,FREQ(33),THETAS(50,33),XOVERD(50,33),NTAB(33),
8      NTEST,XDMAX(33),THETSM(33),RNZDIA,NZTYPE(10),NFREQ
      REAL*8 TEMP(70),TFRQC(70),U(16),C(70),TOL(2),U2(16)
C*****
      REAL*8 PCT,FI
C*****
C
      COMMON /OUTFLO/ IPFRQC,IPANGL,IPFNDM,IPROOT,IPFNDY
      DATA TOL /6.0,4.0/
      IPRINT = 0
      IF(KLISF.NE.0)G00 T0 1000
C
C      BEGIN BY CURVE-FITTING DECIBELS VS FREQUENCY FOR EACH MIKE.
C      OBTAIN NEW Y VALUES (DECIBELS) FOR EACH CURVE AND STORE THEM
C      IN PLACE OF THE OLD UNCORRECTED DECIBEL VALUES FOR EACH MIKE.
C

```



```

      DO 20 J=1,MIKES
C
C      IF( IWTB.EQ.0 .AND. IWTE.EQ.0 ) GO TO 34
C
C      SET UP THE ARRAYS WITH WEIGHTED POINTS
C
      IWTBS1 = IWTB - 1
      DO 25 K=1,IWTBS1
      TEMP(K)=DECB(J,K)
      TFRQC(K)=DFLOAT(K)
25  CONTINUE
      MM = IWTB
C
      DO 28 L = IWTB , NFREQ
      DO 30 M=1,2
      IF(L.GT.IWTE.AND.M.EQ.2) GO TO 30
      TEMP(MM) = DECB(J,L)
      TFRQC(MM) = DFLOAT(L)
      MM = MM+1
30  CONTINUE
28  CONTINUE
      MM = MM-1
      GO TO 40
C
C
C      34 CONTINUE
      MM = 0
C
      DO 35 L = 1 , NFREQ
      MM = MM + 1
      TEMP(MM) = DECB(J,L)
      TFRQC(MM) = DFLOAT(L)
35  CONTINUE
C
      40 CONTINUE
C
      IF (IPFRQC.GT.0) IPRINT = 1
      CALL POLFIT(TFRQC,TEMP,MM,KFIT(1),U,C,IPRINT)
      KONE = KFIT(1)+1
      DO 92 LB=1,KONE
92  U2(LB) = U(LB)
C
C      THROW OUT BAD POINTS (DEVIATION > 5%) AND REFIT WHATS LEFT
C
      MCOUNT=MM
      MINPTS = MM-(MM/3)
      IF (IPFRQC.NE.2) IPRINT = 0
      DO 32 II=1,2
      ICH = 0
C
C      PASS THRU THE LIST OF X,Y PAIRS
C      ZERO OUT THE POINTS WHICH EXCEED THE DEVIATION TOLERANCE
C      BUT DON'T DELETE MORE THAN 6 POINTS
C
      DO 44 IJ=1,MM
      PCT = (TEMP(IJ)-C(IJ))*100.0/C(IJ)
      IF (ABS(PCT).LT.TOL(II)) GO TO 44
      MCOUNT = MCOUNT - 1
      IF (MCOUNT.LE.MINPTS) GO TO 42

```

```

        TEMP(IJ) = 0.0
        TFRQC(IJ) = 0.0
        ICH = 1
44  CONTINUE
        IB=1
42  IF (ICH.EQ.0) GOTO 33
C
C      DELETE THE BAD POINTS & MOVE THE ARRAY CONTENTS UP
C
C
C
        LAST = MM
        KNT = 0
220  KNT = KNT + 1
        IF ( KNT .GT. LAST ) GOTO 225
        IF ( TEMP(KNT) .NE. 0.0 ) GO TO 220
        LAST = LAST - 1
C
        DO 230 KK = KNT , LAST
            TEMP(KK) = TEMP(KK+1)
230  TFRQC(KK) = TFRQC(KK+1)
C
        GOTO 220
225  CONTINUE
        MM = LAST
C
C
        CALL POLFIT (TFRQC,TEMP,LAST,KFIT(1),U,C,IPRINT)
        DO 96 LB=1,KONE
96  U2(LB) = U(LB)
32  CONTINUE
33  DO 60 I=1,NFREQ
        FI = DFL0AT(I)
        DECB(J,I) = POLYX(U2,KFIT(1),FI)
        60 CONTINUE
        20 CONTINUE
1000 CONTINUE
C
C      CALL SUBROUTINES TO CALCULATE NEW PNL & OSPL VALUES
C
        CALL SUBPDB (MIKES,DECB,NFREQ)
        CALL OSPL (DECB,MIKES,NFREQ)
C
C
C      FIT THE CURVE SPL VS. ANGLES FOR BOTH SIDELINEA AND SIDELINEB
C      MIKES.  STORE THE COEFFICIENTS IN COFA & COFB ARRAYS BY COLUMN.
C      REPLACE THE UNCORRECTED DECIBEL VALUES IN SDLNA & SDLNB
C
        K = MIKEA+1
        K2 = KFIT(2)+1
        DO 120 I=1,NFREQ
            IPRINT = 0
            IF (MIKEA.EQ.0) GOTO 155
            IF (IPANGL.EQ.1.OR.IPANGL.EQ.3) IPRINT = 1
            CALL POLFIT(ANGLES, DECB(1,I),MIKEA,KFIT(2),U,C,IPRINT)
            DO 140 J=1,MIKEA
                DECB(J,I) = C(J)
140  CONTINUE
            DO 200 L=1,K2

```

```

      COFA(L,1) = U(L)
200 CONTINUE
155 IF (MIKEB.EQ.0) GOTO 120
      IPRINT = 0
      IF (IPANGL.EQ.2.OR.IPANGL.EQ.3) IPRINT = 1
      CALL POLFIT(ANGLES(K), DECB(K,1),MIKEB,KFIT(2),U,C,IPRINT)
      DO 160 J=1,MIKEB
      J2 = J + MIKEA
      DECB(J2,1) = C(J)
160 CONTINUE
      DO 180 L=1,K2
      COFB(L,1) = U(L)
180 CONTINUE
120 CONTINUE
      CALL SUBPDB (MIKEB,DECB,NFREQ)
      CALL OSPL (DECB,MIKEB,NFREQ)
      RETURN
      END
*DECK POLFIT
      SUBROUTINE POLFIT(X,Y,M,N,U,C,IP)
C
C -----
C
C      PURPOSE:  TO PERFORM A LEAST SQUARES POLYNOMIAL
C      CURVE FIT OF DEGREE N USING M GIVEN POINTS.
C
C      X      THE ARRAY CONTAINING THE X COORDINATES
C      Y      THE ARRAY CONTAINING THE Y COORDINATES
C      M      THE NUMBER OF POINTS TO BE FIT (LIMIT IS 100)
C      N      THE DEGREE OF FIT (LIMIT IS 15)
C      U      ON RETURN THE ARRAY OF N+1 COEFFICIENTS
C      C      ON RETURN THE ARRAY OF CORRECTED Y COORDINATES
C      IP     PRINT FLAG -- 1=ON (RESULTS ARE PRINTED),0=OFF(NO OUTPUT)
C
C      THIS DECK IS INTENDED FOR USE ON THE CDC 7600.  IF THIS PROGRAM
C      IS BEING USED ON TSS/360, REPLACE THE DIMENSION ARRAY1(13),
C      ARRAY2(12) CARD WITH :
C      DOUBLE PRECISION ARRAY1(13),ARRAY2(12)
C
C -----
C
C*****
      REAL*8 X(35),Y(35),Q(100),P(100),C(35)
      REAL*8 A(16),ALPH(12),B(16),S(16),G(16),U(16)
      REAL*8 ARRAY1(13),ARRAY2(12)
C*****
      REAL*8 D,XMEA,YMEA,HH,XMEAN,YMEAN,ERR,YSTDER,E1,F1,
1      W1,W,V,S1,T,T3,T5,ROOT,Q7,P0S,Q8,PCT
C*****
      DATA ALPH/5H A ,5H + B ,5H + C ,5H + D ,5H + E ,5H + F ,
*5H + G ,5H + H ,5H + I ,5H + J ,5H + K ,5H + L /
      DATA ARRAY1/5H ,5H X ,5H X**2,5H X**3,5H X**4,5H X**5,
*5H X**6,5H X**7,5H X**8,5H X**9,5H X**10,5H X**11,5H X**12/
      DATA ARRAY2/5H FIRST,6H SECOND,5H THIRD,6H FOURTH,5H FIFTH,5H SIXTH,
*7H SEVENTH,6H EIGHTH,5H NINTH,5H TENTH,6H ELEVENTH,7H TWELFTH/
      D=DFLOAT(M)
      G(1)=0.0
      N=N+1
      IF (N.GT.12) GO TO 230
      IF (M.LT.N) GO TO 240

```

```

XMEA=0.0
YMEA=0.0
HUH=0
DO 10 I=1,M
XMEA=XMEA+X(I)
YMEA=YMEA+Y(I)
10 HUH=HUH+Y(I)**2
XMEAN=XMEA/D
YMEAN=YMEA/D
ERR=(D*HUH-YMEA**2)/(D**2-D)
IF (IP.EQ.0) GOTO 500
YSTDER=DSQRT(ERR)
WRITE (6,250) M
WRITE (6,260) XMEAN,YMEAN
WRITE (6,270) YSTDER
500 DO 20 I=1,M
P(I)=0.0
20 Q(I)=1.0
DO 30 I=1,11
A(I)=0.0
B(I)=0.0
30 S(I)=0.0
E1=0.0
F1=0.0
W1=0
N4=12
I=1
40 W=0.0
DO 50 L=1,M
50 W=W+Y(L)*Q(L)
S(I)=W/W1
IF (I-N4.GE.0) GO TO 80
IF (I-M.GE.0) GO TO 80
E1=0.0
DO 60 L=1,M
60 E1=E1+X(L)*Q(L)*Q(L)
E1=E1/W1
A(I+1)=E1
W=0.0
DO 70 L=1,M
70 V=(X(L)-E1)*Q(L)-F1*P(L)
P(L)=Q(L)
Q(L)=V
W=W+V*V
F1=W/W1
B(I+2)=F1
W1=W
I=I+1
GO TO 40
80 DO 90 L=3,12
90 G(L)=0.0
G(2)=1.0
LL=2
DO 130 J=1,N
S1=0.0
DO 110 L=1,N
IF (L.EQ.1) GO TO 100
LL=L+1
G(LL)=G(LL)-A(L)*G(LL-1)-B(L)*G(LL-2)
100 S1=S1+S(L)*G(LL)

```

```

110 CONTINUE
    U(J)=S1
    L=N+1
    DO 120 I2=2,N
        G(L)=G(L-1)
120  L=L-1
    LL=2
130  G(2)=0.0
    T=0.0
    DO 150 L=1,M
        C(L)=0.0
        J=N
        DO 140 I2=1,N
            C(L)=C(L)*X(L)+U(J)
140  J=J-1
        T3=Y(L)-C(L)
150  T=T+T3**2
        IF (M.NE.N) GO TO 160
        T5=0.0
        GO TO 170
160  T5=T/(D-DFLOAT(N))
        ROOT=DSQRT(T5)
        IF (IP.EQ.1) WRITE(6,280) ROOT
170  IF (DABS(ERR).LT.0.00001) ERR = 0.001
        Q7 = 1.0-T/(ERR*(D-1.0))
        LESS=N-1
        IF (IP.EQ.0) GO TO 501
        WRITE (6,290) LESS
        WRITE (6,300) Q7
        WRITE (6,310)
        DO 180 J=1,N
            I2=J-1
180  WRITE (6,320) I2,ALPH(J),U(J)
        N1 = N-1
        WRITE(6,330) ARRAY2(N1)
        N1 = N
        IF (N1.GT.9) N1=9
        WRITE(6,340) ALPH(1),ALPH(2),ARRAY1(2),(ALPH(I),ARRAY1(I),I=3,N1)
        IF (N.LE.9) GO TO 191
        WRITE(6,350) (ALPH(I),ARRAY1(I),I=10,N)
191  WRITE(6,360)
501  KOUNT = 0
        DO 220 L=1,M
            POS=DABS(Y(L)-C(L))
            IF (POS.LT.0.1E-08) Y(L)=C(L)
            Q8=Y(L)-C(L)
            IF (C(L).EQ.0.0) GO TO 200
            PCT=100.0*Q8/C(L)
            IF (IP.EQ.1) WRITE(6,370) X(L),Y(L),C(L),Q8,PCT
            GO TO 210
200  IF (IP.EQ.1) WRITE(6,380) X(L),Y(L),C(L),Q8
210  KOUNT=KOUNT+1
        IF (KOUNT.LT.42) GO TO 220
        IF (IP.EQ.1) WRITE(6,360)
        IF (KOUNT.EQ.42) KOUNT=0
220  CONTINUE
        N=N-1
        RETURN
230  WRITE (6,390)
        STOP

```

```

240 LESS=N-1
    WRITE (6,400) LESS
    RETURN
C
250 FORMAT (1H1,6X,18HNUMBER OF POINTS =,14)
260 FORMAT (7X,17HMEAN VALUE OF X =,F10.4 / 7X,17HMEAN VALUE OF Y =,
-      F10.4)
270 FORMAT (7X,21HSTANDARD ERROR OF Y =,F10.4)
280 FORMAT (1H0,6X,35HSTANDARD ERROR OF ESTIMATE FOR Y =,F9.4 //)
290 FORMAT (7X,16HDEGREE OF FIT = ,12)
300 FORMAT (7X,23HINDEX OF DETERMINATE = ,F9.7 /)
310 FORMAT (7X,4HTERM,4X,6HLETTER,10X,11HCoefficient/)
320 FORMAT (8X,12,6X,A3,7X,E23.16)
330 FORMAT (1H0,6X,12HEQUATION IS ,A8,18H DEGREE POLYNOMIAL /)
340 FORMAT (7X,3HY =,2A3,A1,1X,7(A3,A4,1X))
350 FORMAT (//1X,3(3H + ,A2,A5//))
360 FORMAT (1H1,9X,8HX-ACTUAL,7X,8HY-ACTUAL,8X,6HY-CALC,5X,
-      10HDIFFERENCE,5X,8HPCT-DIFF /)
370 FORMAT (7X,F12.6,3X,F12.6,3X,F12.6,3X,F10.6,5X,F8.4)
380 FORMAT (7X,F12.6,3X,F12.6,3X,F12.6,3X,F10.6,5X,8HINFINITE)
390 FORMAT (1X,38HERROR-----ELEVENTH DEGREE IS THE LIMIT)
400 FORMAT (1X,44HERROR-----TOO FEW POINTS FOR FITTING DEGREE,13)
    END

```

APPENDIX B

NOISE 4

```

CPROGRAMME SOURCE N04BD
C
C-----
C
C    MAIN PROGRAM (DRIVER) FOR PHASE 2 OF
C    ACOUSTICS DATA ANALYSIS.
C
C    LOGICAL UNIT ASSIGNMENTS -
C
C        05 = USER PROGRAM INPUTS
C        06 = OUTPUT (LINE PRINTER) FILE
C        07 = BINARY DUMP FROM PHASE 1 ANALYSIS (SAVED IN DISK FILE)
C        9  = ECRITURE SUR RNOISE DE PSIVAL-XDVAL
C        08 = CORRECTION DATA
C        10 = NEAR FIELD SPL READINGS AND MIKE ANGLES
C        69 = TRACES : MESURES-POLYNOMES-DIRECTIVITE CHAMP LOINTAIN
C
C-----
C
C    REAL*8 DUMP,DUMP1,DUMP2,DUMP3
C    REAL*8 PRESS,RTOK,FTOM,TEMPK,ABDBM,DISTME
C    REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
C    REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
C    *TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
C    *COFNR,AN,YY,R2D,D2R,XDIS
C    COMMON /SUB1/ VJ, FREQ(33),THETAS(50,33),XOVERD(50,33),
C    * NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
C    * NZTYPE(10),NFREQ
C
C    COMMON /SUB2/ PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
C    1 FFSPL(50,35),ST(35),STP(35),
C    3 IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
C    4 SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
C    5 ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
C    6 PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
C    7 R2D,D2R, XDIS(70,2),IMRC,IBNC,IRC,IAAC,
C    8 IPRQC,IPANGL,ICALL,NZTYPE
C
C    COMMON/TST/ITEST(70,30),N0FR(30)
C    COMMON/TRAFLG/KTRAC,KTFFSPL,KTDECBM,KTDECBP
C    COMMON/IDENT/NP01,KMIC,NP0IF,KLI0,IGLDEB,IGLFIN
C    COMMON/TITRE/LIB(10),IDATE(3),DB1(8,4),IPI(8),LIB1(20),LIB2(20)
C    DIMENSION ITIT(20),XMOD(44)
C    DIMENSION DUMP(3446),DUMP1(74),DUMP2(1800),DUMP3(132),LIBID
C    *(10),IDATBID(3)
C    EQUIVALENCE (DUMP(1),VJ)
C    EQUIVALENCE (DUMP1(1),ST(1))
C    EQUIVALENCE (DUMP2(1),PSIVAL(1))
C
C    DATA PRESS,RTOK,FTOM,D2R /1.0, 0.55555556, 0.3048, 0.017453293/
C*****CODE DE TRAITEMENT
C    KODT 0 TRAITEMENT BANDE MODANE
C    KODT 1 TRAITEMENT DONNEES AMES
C    KODT 2 TRAITEMENT DONNEES AMES BANDES
C    READ(5,101)KODT
C    101 FORMAT(I5)
C*****MISE A ZERO DU TABLEAU DECB2
C    DO 10 I=1,70
C    DO 10 J=1,30
C    10 DECB2(I,J)=0.

```



```

      R2D = 1.0/D2R
C
C
C      READ UNFORMATTED DATA FROM PHASE 1 PROGRAM
C
      NW=7
      READ(NW,END=92)DUMP
      READ(NW,END=92)DUMP3
      READ(NW,END=92)NBID
      READ(NW,END=92)LIBID
      READ(NW,END=92)IDATBID
      READ(NW,END=92)LIB1
      READ(NW,END=92)LIB2
      GO TO 100
92  WRITE(6,91);STOP
91  FORMAT (1H1,64HEOF ENCOUNTERED WHILE READING DATA FROM PHASE 1, LO
      *GICAL UNIT 07 /)
C
C
100  CALL READIN(LIST,KPF,KODT,ITIT)
      IF(KPF.EQ.0)GO TO 300
      CALL STEST(MIKEA,NFREQ,ANGLES,FREQ)
300  CONTINUE
C***** PREPARATION DU TRACE
      IF(KTRAC.EQ.0) GO TO 150
      DO 160 I=1,44
      XM0D(I)=0.
160  CONTINUE
      KPLANC=0
      CALL OPENTR(69,XM0D)
150  CONTINUE
      CALL OSPL(DECB2,MIKEA,NFREQ)
      CALL SUBPDB (MIKEA,DECB2,NFREQ)
      WRITE(6,1001)
1001 FORMAT(1H1,30X,55HSOUND PRESSURE LEVELS READ AS INPUTS FOR THIS EX
      *ECUTION /)
      CALL OUTPUT (XDIS,DECB2,ANGLES,FREQ,MIKEA,NFREQ,HTMIKE)
      IF(LIST.EQ.1)GO TO 201
C
      IF (IMRC.NE.1) GOTO 40
      ICALL = 1
      CALL CORECT
40  IF (IBNC.NE.1) GOTO 50
      ICALL = 2
      CALL CORECT
50  IF (IRC.NE.1) GOTO 60
      ICALL = 3
60  IF (IAAC.NE.1) GOTO 80
C
C
C      ATMOSPHERIC ABSORPTION CORRECTIONS
C      PRESS = ATMOSPHERES
C      RTOK = RANKIN TO KELVIN CONV FACTOR
C      FTOM = FEET TO METERS CONV FACTOR
C
      TEMPK = TTEMP*RTOK
      DO 200 NF=1,NFREQ
      CALL AIFAB (PRESS,TEMPK,THUM ,FREQ(NF),ABDBM)
      DO 220 MIC=1,MIKEA
      DISTME = XDIS(MIC,2)*FTOM

```

```

      DECB2(MIC,NF) = DECB2(MIC,NF) + ABDBM*DISTME
220 CONTINUE
200 CONTINUE
C
  80 IF(1MRC.EQ.0.AND.1BNC.EQ.0.AND.1RC.EQ.0.AND.1AAC.EQ.0)GOTO 90
      CALL OSPL (DECB2,MIKEA,NFREQ)
      CALL SUBPDB(MIKEA,DECB2,NFREQ)
      WRITE(6,1002)
1002 FORMAT(1H1,30X,22HDATA AFTER CORRECTIONS /)
      CALL OUTPUT (XDIS,DECB2,ANGLES,FREQ,MIKEA,NFREQ,HTMIKE)
C
  90 CONTINUE
      CALL THMERG
      CALL COMP2
      CALL NEWPSI
C*****
C***** TRACE DES MESURES
      IF(KTDECBM.EQ.0) GO TO 190
      NCAR=17
      CALL TDECB1(KPLANC,DECB2,ANGLES,FREQ,NFREQ,MIKEA,
1'FIG.      MESURES. ',NCAR,XMOD)
190 CONTINUE
      CALL CURVFT
C***** TRACE DES POLYNOMES D INTERPOLATION
      IF(KTDECBP.EQ.0) GO TO 180
      NCAR=35
      CALL TDECB1(KPLANC,DECB2,ANGLES,FREQ,NFREQ,MIKEA,
1'FIG.      POLYNOMES D INTERPOLATION. ',NCAR,XMOD)
180 CONTINUE
C
      WRITE(6,1003)
1003 FORMAT(1H1,30X,20HDATA AFTER SMOOTHING / )
      CALL OUTPUT(XDIS,DECB2,ANGLES,FREQ,MIKEA,NFREQ,HTMIKE )
      CALL OSPL(DECB2,MIKEA,NFREQ )
      CALL SUBPDB (MIKEA,DECB2,NFREQ)
C
C
C
      CALL SPCTRA
C*****TRACE DU CHAMP LOINTAIN
      IF(KTFFSPL.EQ.0)GO TO 170
      NCAR=25
      MIKEA=50
      CALL OSPL1(FFSPL,MIKEA,NFREQ)
      CALL TDECB2(KPLANC,FFSPL,PSI2,PSIVAL,FREQ,NFREQ,MIKEA,YY,
*FIG.      CHAMP LOINTAIN. ',NCAR,XMOD)
170 CONTINUE
C
C
C
      ECRITURE SUR DISQUE (TRACE PSI-X2 ECH NASA)
      WRITE(9)DUMP1
      WRITE(9)DUMP2
      WRITE(9)NP01
      WRITE(9)LIB
      WRITE(9)IDATE
      REWIND 9
C
C***** FIN DE TRACE
      IF(KTRAC.EQ.0) STOP
      CALL CLOSTR(XMOD)

```

```

WRITE(6,1004) KPLANC
1004 FORMAT(1H1///3X,'NOMBRE DE PLANCHES=',I3///)
C*****
201 CONTINUE
STOP
END
*DECK AIFAB
SUBROUTINE AIFAB (P,T,RH,CF,ABDBM)
C
C -----
C THIS PROGRAM CALCULATES THE ABSORPTION OF SOUND IN AIR AS A
C FUNCTION OF TEMPERATURE, HUMIDITY, PRESSURE AND FREQUENCY.
C THE PROGRAM SHOULD NOT BE USED FOR CALCULATIONS OUTSIDE
C THE TEMPERATURE RANGE OF 0 DEG F (-20 DEG C) THROUGH 104 DEG F
C (40 DEG C)
C
C ABDBM = AMPLITUDE ABSORPTION COEFFICIENT IN DB/METER
C ABDBSC = AMPLITUDE ABSORPTION COEFFICIENT IN DB/SECONDS
C ABDBTF = AMPLITUDE ABSORPTION COEFFICIENT IN DB/1000FT
C ABDAM = AMPLITUDE ABSORPTION COEFFICIENT PER WAVELENGTH
C ALPHA = AMPLITUDE ABSORPTION COEFFICIENT IN NEPERS PER METER
C
C CF = ACOUSTIC FREQUENCY
C P = AMBIENT PRESSURE IN ATM
C PS = SATURATED VAPOR PRESSURE FOR WATER IN ATMOSPHERES
C T = TEMPERATURE IN DEGREES KELVIN
C TC = TEMPERATURE IN DEGREES CENTIGRADE
C TF = TEMPERATURE IN DEGREES FAHRENHEIT
C WAVEL = WAVELENGTH OF SOUND WAVE
C -----
C
REAL*8 ABDBM,CF,P,RH,T
REAL*8 P1,T1,TC,TF,VEL,VELFPS,T01,PS,H,FR02,FRN2,ALPHA,WAVEL,
*ABDAM,ABDBTF,ABDBSC
P1 = 3.14159
T1 = T/293.
TC = T-273.
TF = TC*1.8+32.
VEL = 343.4*DSQRT(T1)
VELFPS = VEL*3.28
T01 = 273.16
C
PS = 10.79586*(1.-T01/T)-5.02808*DL0G10(T/T01)+1.50474E-4*(1.-10. -
C**(-8.29692*((T/T01)-1.)))+0.42873E-3*(10.**((4.76955* -
C(1.-(T01/T)))-1.))-2.2195983
PS = 10.**PS
H = PS/P*RH
FR02 = P*(24.+4.41E04*H*(0.05+H)/(0.391+H))
FRN2 = P/DSQRT(T1)*(9.+350.*H*DEXP(-6.142*((1./T1)**.331-1.)))
ALPHA = 1.84E-11+2.1913E-4/T1*P*(2239.1/T)**2*DEXP(-2239.1/T) -
C /(FR02+(CF**2/FR02))
ALPHA = ALPHA+8.1619E-4/T1*P*(3352./T)**2*DEXP(-3352./T) -
C /(FRN2+(CF**2/FRN2))
ALPHA = ALPHA*DSQRT(T1)*CF**2/P
WAVEL = VEL/CF
ABDAM = ALPHA*WAVEL
ABDBTF = ALPHA*2647.
ABDBM = ALPHA*8.6860
ABDBSC = ALPHA*VEL*8.686
CONTINUE

```

```

      RETURN
      END
*DECK ATMAT
      SUBROUTINE ATMAT(T,RH,DIST,FREQ,ATT)
C
C -----
C
C   ATMAT STANDS FOR ATMOSPHERIC ATTENUATION
C
C   COMPUTES EXCESS ATMOSPHERIC ATTENUATION IN DECIBELS FOR GIVEN
C   TEMPERATURE, RELATIVE HUMIDITY, DISTANCE, AND FREQUENCY.
C   USES EMPIRICAL CURVE FITS OF DATA CONTAINED IN SOCIETY OF
C   AUTOMOTIVE ENGINEERS AEROSPACE RECOMMENDED PRACTICE NO. 266,
C   AUGUST, 1964
C
C   T      TEMPERATURE (DEGREES FAHRENHEIT)
C   RH      RELATIVE HUMIDITY
C   DIST    DISTANCE(FEET)
C   FREQ    FREQUENCY (HERTZ)
C   ATT     ATTENUATION (DECIBELS)
C
C -----
C
      REAL*8 ATT,DIST,FREQ,RH,T
      REAL*8 A,AC,AMM,HA,HMM,HH,AA,HTEST
      DIMENSION A(22)
      DATA A/0.870,0.750,0.652,0.570,0.505,0.452,0.406,0.369,0.335,
10.308,0.286,0.268,0.253,0.240,0.231,0.225,0.220,0.215,0.210,
20.208,0.202,0.200/
C
      AC = (0.1*(FREQ/1000.0)**2.05)/(1.651-.00103*T)**2.05
      AMM = (10.0*(FREQ/1000.0)**1.003)/10.0**((0.52-.00504*(T+
1DSQRT(256.0-(10.0-T/5.0)**2)))
      HA = 0.25 * RH/10.0**((1.493-.01638*T-.02*DSQRT(128.2 -
1(10.0-T/5.0)**2))
      HMM = 10.0**((0.4973*LOG10(FREQ)-1.4894)
      HH = HA / HMM
      IF(HH.GT.0.25) GO TO 1
      AA = 1.2 * HH
      GO TO 8
1 IF(HH.GT.0.60) GO TO 2
      AA = 1.543 * HH - .086
      GO TO 8
2 IF(HH.GT.0.95) GO TO 3
      AA = 0.84 + 0.16 * DSIN(3.14159/2.0*(HH-0.6) / 0.35)
      GO TO 8
3 IF(HH.GT.1.25) GO TO 4
      AA = 0.87 + 0.13 * DCOS(3.14159/2.0*(HH-0.95) / 0.3)
      GO TO 8
4 IF(HH.GT.6.5) GO TO 7
      HTEST = 1.25
      DO 5 I = 2,22
      HTEST = HTEST + 0.25
      IF(HH.LE.HTEST) GO TO 6
5 CONTINUE
6 AA = A(I) + ((HTEST-HH) / 0.25) * (A(I-1)-A(I))
      GO TO 8
7 AA = 0.2
8 CONTINUE
      ATT = (AMM*AA+AC)*(DIST*0.001)

```

```

      RETURN
      END
*DECK COMP2
      SUBROUTINE COMP2
C
      REAL*8 C,TRM,F1,VDIF,F2,TH1,TH2,D
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
      *TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
      *COFNR,AN,YY,R2D,D2R,XDIS
      COMMON /SUB1/ VJ,          FREQ(33),THETAS(50,33),XOVERD(50,33),
      *          NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
      *          NZTYPE(10),NFREQ
C
      COMMON /SUB2/ PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
      1          FFSPL(50,35),ST(35),STP(35),
      3          IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
      4          SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
      5          ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
      6          PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
      7          R2D,D2R,          XDIS(70,2),IMRC,IBNC,IRC,IAAC,
      8          IPFRQC,IPANGL,ICALL,NZTYPE
C
C
C      DATA STATEMENT DEFINES ACOUSTIC VELOCITY, C IN FEET PER SEC
      DATA C /1115.0/
C
      TRM = VAMB/C
      F1 = RNZDIA/VJET
      VDIF = VJET - VAMB
      IF ( VDIF .EQ. 0.0 ) VDIF = VJET
      F2 = RNZDIA/VDIF
C
      DO 1000 I = 1 , NFREQ
C
      ST(I) = FREQ(I) * F1
      STP(I) = FREQ(I) * F2
C
      NJ = NTAB(I)
      IF (NJ.EQ.0) GO TO 1000
C
C      TRANSFORM EACH THETA-S TO PSI-S FOR THIS FREQUENCY
C
      DO 500 J = 1 , NJ
      TH1 = DSIN(D2R * (THETAS(J,I) - 90.)) + TRM
      TH2 = DCOS(D2R * (THETAS(J,I) - 90.))
      IF ( TH1 .EQ.0.0 .OR. TH2 .EQ. 0.0 ) GO TO 450
      PSIS(J,I) = DATAN(TH1/TH2)*R2D + 90.
      GO TO 500
450 CONTINUE
      PSIS(J,I) = 0.0
500 CONTINUE
C
C      FOR EACH NEW PSIS OF THIS FREQUENCY, CONVERT X1 TO X2
C      (XOVERD TO XDN) FOR EACH NEW PSIS VALUE
C
      D = 0.0
      DO 600 J=1,NJ
      IF (PSIS(J,I).EQ.0.0) GO TO 650
      CALL TAINI (THETAS(1,I),XOVERD(1,I),PSIS(J,I),XDN(J,I),NJ,3,NER,D)

```

```

        IF (NER.NE.1) GOTO 650
        GOTO 600
650 XDN(J,1) = 0.0
600 CONTINUE
C
C 1000 CONTINUE
C
        RETURN
        END
*DECK CORECT
SUBROUTINE CORECT
C
C      ICALL = 1  FOR MIKE RESPONSE CORRECTIONS
C      ICALL = 2  FOR REVERBERATION CORRECTIONS
C      ICALL = 3  FOR BACKGROUND NOISE CORRECTIONS
C
C
      REAL*8 CORVAL,DUMMY1,DUMMY2
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
      *TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
      *COFNR,AN,YY,R2D,D2R,XDIS
      COMMON /SUB1/ VJ,          FREQ(33),THETAS(50,33),XOVERD(50,33),
      *          NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
      *          NZTYPE(10),NFREQ
C
      COMMON /SUB2/  PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
      1  FFSPL(50,35),ST(35),STP(35),
      3  IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
      4  SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
      5  ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
      6  PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
      7  R2D,D2R,          XDIS(70,2),IMRC,IBNC,IRC,IAAC,
      8  IPFRQC,IPANGL,ICALL,NZTYPE
C
      DIMENSION CORR(50,33)
C
      EQUIVALENCE (CORR(1,1),XOVERD(1,1))
C      EQUIVALENCE STATEMENT TO SAVE CORE SPACE
C
      MIKEB = 0
C
      INITIALIZE ARRAY OF CORRECTION VALUES
C
      DO 100 J=1,33
      DO 100 K=1,50
100 CORR(K,J) = 0.0
C
      REWIND 8
      GOTO (200,400,600),ICALL
C
C
C      THIS SECTION IS USED FOR MIKE RESPONSE CORRECTIONS
C
C
200 READ (8) IRUN,ICHAN,IBAND,CORVAL,DUMMY1,DUMMY2
      IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GOTO 300
      IF (ICHAN.LT.1.OR.ICHAN.GT.50) GOTO 200
      IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GOTO 200
      CORR(ICHAN,IBAND) = CORVAL

```

```

      GOTO 200
C
300 WRITE(6,801)
801 FORMAT (1H1,10X,31HMIKE RESPONSE CORRECTION VALUES /)
      CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
C
C      APPLY CORRECTIONS
C
250 DO 225 J=1,NFREQ
      DO 225 K=1,MIKEA
225  DECB2(K,J) = DECB2(K,J)-CORR(K,J)
      GOTO 900
C
C
C      THIS SECTION APPLIES REVERBERATION CORRECTIONS
C
400 READ (8) IRUN,ICHAN,IBAND,DUMMY1,CORVAL,DUMMY2
      IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GOTO 500
      IF (ICHAN.LT.1.OR.ICHAN.GT.50) GOTO 400
      IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GOTO 400
      CORR(ICHAN,IBAND) = CORVAL
      GOTO 400
C
500 WRITE(6,802)
802 FORMAT (1H1,10X,31HREVERBERATION CORRECTION VALUES /)
      CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
      GOTO 250
C
C
C      APPLY BACKGROUND NOISE CORRECTIONS HERE
C
600 READ (8) IRUN,ICHAN,IBAND,DUMMY1,DUMMY2,CORVAL
      IF (IRUN.EQ.0.AND.ICHAN.EQ.0) GOTO 700
      IF (ICHAN.LT.1.OR.ICHAN.GT.50) GOTO 600
      IF (IBAND.LT.1.OR.IBAND.GT.NFREQ) GOTO 600
      CORR(ICHAN,IBAND) = CORVAL
      GOTO 600
C
700 WRITE (6,803)
803 FORMAT (1H1,10X,34HBACKGROUND NOISE CORRECTION VALUES /)
      CALL OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
      GOTO 250
C
C      SUBROUTINE EXIT
C
900 RETURN
      END
*DECK CURVFT
      SUBROUTINE CURVFT
C
      REAL*8 TEMP,XCNT,COEF,YNEW,TOL,PCT,FJ
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
      *TTEMP,THUM,PSI10,PSI11,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
      *COFNR,AN,YY,R2D,D2R,XDIS
      COMMON /SUB1/ VJ, FREQ(33),THETAS(50,33),XOVERD(50,33),
      * NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
      * NZTYPE(10),NFREQ

```

```

C      COMMON /SUB2/  PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
1      FFSPL(50,35),ST(35),STP(35),
3      IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
4      SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
5      ANGLS(70),DECB2(70,35),HTSRCE,HTMIKE(70),
6      PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
7      R2D,D2R,      XDIS(70,2),IMRC,IBNC,IRC,IAAC,
8      IPFRQC,IPANGL,ICALL,NTYPE

C      DIMENSION  TEMP(65),XCNT(65),COEF(11),YNEW(65)

C      DATA TOL /6.0/

C      IPRINT = 0

C
C      FIRST CURVE FIT (SMOOTH DATA) FOR EACH ANGLE ACROSS
C      BAND NO. VS SPL.  USE THE WIEGHTING PARAMS ENTERED BY
C      THE USER.  DO NOT SAVE THE COEFFICIENTS FOR THIS PASS.
C
C      IF (IWT1.EQ.0) IWT1=NFREQ
C      DO 200 MIC = 1,MIKEA
C
C      DO 100 K=1,IWT1
C      XCNT(K) = DFL0AT(K)
100  TEMP(K) = DECB2(MIC,K)
C
C      NEXT = IWT1+1
C      LAST = IWT2
C      KNT = IWT1
C      IF (IWT1.EQ.NFREQ) GOTO 108
C
C      DO 102 K=NEXT, LAST
C      DO 103 L=1,2
C      KNT = KNT+1
C      XCNT(KNT) = DFL0AT(KNT)
C      TEMP(KNT) = DECB2(MIC,K)
103  CONTINUE
102  CONTINUE
C
C      DO 104 K=IWT2,NFREQ
C      KNT = KNT+1
C      XCNT(KNT) = DFL0AT(KNT)
C      TEMP(KNT) = DECB2(MIC,K)
104  CONTINUE
C
C
C      108 IF (IPFRQC.GT.0) IPRINT=1
C      CALL POLFIT (XCNT,TEMP,KNT,KFIT(1),COEF,YNEW,IPRINT)
C
C
C      DELETE BAD PTS (.GT.TOL)  .. BUT NOT MORE THAN 1/5-TH OF THEM
C      MOVE ARRAY CONTENTS UP AND REFIT DATA
C
C      MINPTS = KNT-(KNT/5)
C      MSUB = KNT-MINPTS
C      MCNT = 0

```



```

C      DO 44 IJ = 1,KNT
      PCT = ((TEMP(IJ)-YNEW(IJ))/YNEW(IJ))*100.0
      IF (DABS(PCT).LT.TOL) GOTO 44
      IF (MCNT.GT.MSUB) GOTO 44
      TEMP(IJ) = 0.0
44    CONTINUE
C
      LAST = KNT
      K = 0
220   K = K+1
      IF (K.GT.LAST) GOTO 225
      IF (TEMP(K).NE.0.0) GOTO 220
      LAST = LAST+1
C
      DO 230 KK=K,LAST
      TEMP(KK) = TEMP(KK+1)
230   XCNT(KK) = XCNT(KK+1)
      TEMP(LAST+1) = 0.
      XCNT(LAST+1) = 0.
      GOTO 220
225   IPRINT=0
      IF (IPFRQC.EQ.2) IPRINT=1
      CALL POLFIT (XCNT,TEMP,LAST,KFIT(1),COEF,YNEW,IPRINT)
C
C
      DO 120 J=1,NFREQ
      FJ = DFLOAT(J)
120   DECB2(MIC,J) = POLYX(COEF,KFIT(1),FJ)
C
200   CONTINUE
C
C
C      NOW CURVE FIT ANGLES VS DECIBELS FOR EACH FREQUENCY
C      AND SAVE THE KFIT(2)+1 COEFFICIENTS IN ARRAY COFNR FOR
C      EACH CURVE FIT PERFORMED
C
C
      IPRINT = 0
      IF (IPANGL.NE.0) IPRINT=1
      DO 390 K=1,MIKEA
390   XCNT(K) = ANGLES(K)
C
      DO 400 NF = 1,NFREQ
C
      CALL POLFIT (XCNT,DECB2(1,NF),MIKEA,KFIT(2),COFNR(1,NF),
      *          YNEW,IPRINT)
C
      DO 450 K=1,MIKEA
450   DECB2(K,NF) = YNEW(K)
400   CONTINUE
C
      RETURN
      END
*DECK NEWPSI
      SUBROUTINE NEWPSI
C
      REAL*8 PSIINC,FN,DMON
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,

```

```

*TEMP, THUM, PSIL0, PSIH1, ANGLES, DECB2, HTSRCE, HTMIKE, PSIVAL, XDVAL,
*COFNR, AN, YY, R2D, D2R, XDIS
COMMON /SUB1/ VJ,      FREQ(33), THETAS(50,33), XOVERD(50,33),
*
*      NTAB(33), NTEST, XDMAX(33), THETSM(33), RNZDIA,
*      NZTYPE(10), NFREQ
COMMON /SUB2/ PSIS(50,35), XDN(50,35), PSI1(50,35), PSI2(50,35),
1      FFSPL(50,35), ST(35), STP(35),
3      IWT1, IWT2, VAMB, VJET, KFIT(2), SDHUM,
4      SDTEMP, TTEMP, THUM, PSIL0, PSIH1, MIKEA, IBIDON,
5      ANGLES(70), DECB2(70,35), HTSRCE, HTMIKE(70),
6      PSIVAL(50), XDVAL(50,35), COFNR(11,35), AN, YY,
7      R2D, D2R,      XDIS(70,2), IMRC, IBNC, IRC, IAAC,
8      IPFRQC, IPANGL, ICALL, NTYPE
C
C      K=3
C      PSIINC = (PSIH1-PSIL0)/50.0
C
C      PSIVAL(1) = PSIL0
C      KNT = 2
C      FN = 1.0
C
C      120 PSIVAL(KNT) = FN*PSIINC+PSIL0
C      KNT = KNT+1
C      IF (KNT.GT.50) GOT0 100
C      FN = FN+1.0
C      GOT0 120
C
C
C      100 D0 200 IC0L=1, NFREQ
C      DM0N = 0.0
C      ICNT = NTAB(IC0L)
C      IF (ICNT.EQ.0) GOT0 200
C
C      D0 300 IROW = 1,50
C      CALL TAIN (PSIS(1,IC0L), XDN(1,IC0L), PSIVAL(IROW),
*      XDVAL(IROW,IC0L), ICNT, K, NER, DM0N)
C      IF (NER.EQ.1) GOT0 300
C      XDVAL(IROW,IC0L) = 0.0
C      300 CONTINUE
C
C      200 CONTINUE
C
C      RETURN
C      END
*DECK NFCORR
C
C      SUBROUTINE NFCORR ( NTYPE,SLD,R,F,DV,DELD0B )
C
C
C      SUBROUTINE NFCORR IS USED TO COMPUTE NEAR FIELD CORRECTION VALUES FOR
C      PHASE 2 EXTRAPOLATIONS TO THE FAR FIELD.
C      THE CORRECTION VALUES ARE COMPUTED FROM TABLE INTERPOLATIONS USING
C      SUBROUTINE TAIN.
C      FOUR TABLES OF VALUES ARE PROVIDED , ONE FOR EACH NOZZLE TYPE AND
C      SIDELINE DISTANCE.
C      IF Y VALUES ARE TO BE EXTRAPOLATED , THEY(X0UT) WILL BE SET TO 0.0
C
C      NTYPE = 1      FOR      VFE NOZZLE
C      NTYPE = 2      FOR      STOVEPIPE NOZZLE

```

```

C      NTYPE = 3      FOR      104 TUBE NOZZLE
C      NTYPE = 4      FOR      104 TUBE NOZZLE WITH SHROUD NOZZLE
C
C      THE VALUES 0., 1.3 OR 1.19 WILL BE SUBTRACTED FROM THE INTERPOLATED DELDB ,
C      DEPENDING ON THE SLD VALUE .
C      IF THE SLD IS NOT WITHIN 10 PERCENT OF 4.86, 4.49, 8.5 OR 7.86 , DELDB WILL
C      BE SET EQUAL TO 0.0
C
C      REAL*8 DELDB, DV, F, R, SLD
C      REAL*8 SUBTR, SLD4, SLD1, DIF1, DIF2, PERC1, PERC2, A, SUB, XOUT, XIN, D
C      REAL*8 X1, X, X2, X3, X4
C      REAL*8 Y1, Y, Y2, Y3, Y4
C      COMMON/TBX/X1(25), X2(25), X3(25), X4(25)
C      COMMON/TBY/Y1(25), Y2(25), Y3(25), Y4(25)
C      DIMENSION X(25,4), Y(25,4)
C      DIMENSION SUBTR(4), SLD4(4), KTAB(4), SLD1(4)
C      EQUIVALENCE ( X(1,1) , X1(1)), (Y(1,1), Y1(1))
C
C      DATA X1/.3, .4, .5, .6, .7, .8, .9, 1., 1.5, 1.9, 2., 3., 4., 5., 7., 9., 11., 12.,
2      7*0.0/
C      DATA Y1/2*0., .0001, .9, 3.15, 4.65, 2*4.9, 4.1, 3.75, 3.65, 2.65, 2.05,
2      1.45, .8, .25, 9*0.0/
C      DATA X2/.4, .5, .6, .7, .8, .9, 1., 1.4, 1.5, 1.6, 1.9, 2., 3., 4., 5., 6., 7.,
2      10., 16., 20., 5*0.0/
C      DATA Y2/2*0., .08, 1.05, 2.38, 3.68, 4.1, 4.58, 4.68, 4.65, 4.58, 4.48, 3.8,
2      3.12, 2.83, 2.12, 1.75, .7, 7*0.0/
C      DATA X3/.4, .5, .6, .7, .8, .9, 1., 1.4, 1.6, 1.8, 2., 3., 3.5, 4., 5., 6., 7., 8.,
2      10., 14., 20., 29., 50., 2*0.0/
C      DATA Y3/2*0., .5, 1.28, 1.88, 2.3, 2.75, 3.3, 3.5, 3.8, 3.8, 4.15, 4.2, 4.15,
2      4.03, 3.9, 3.66, 3.45, 2.9, 1.83, .7, 4*0.0/
C      DATA X4/.5, .7, .8, .9, 1., 1.2, 1.4, 1.6, 2., 3., 4., 5., 6., 7., 8., 10., 15.,
2      20., 30., 40., 50., 58., 60., 2*0.0/
C      DATA Y4/2*0., .9, 1.68, 2.13, 3., 3.48, 4., 4.48, 5.05, 5.2, 5.3, 5.25, 5.2,
2      5.19, 4.95, 4.48, 4.0, 3.0, 2.05, .9, 4*0.0/
C
C      DATA KTAB/18, 20, 23, 23/
C      DATA SUBTR/1.3, 1.19, 1.19, 1.19/
C      DATA SLD1 / 4.86, 3*4.49 /
C      DATA SLD4/8.5, 7.86, 7.86, 7.86/
C
C      DIF1 = DABS ( SLD - SLD1(NTYPE) )
C      DIF2 = DABS ( SLD - SLD4(NTYPE) )
C      PERC1 = SLD1(NTYPE)/10.
C      PERC2 = SLD4(NTYPE)/10.
C      IF ( DIF1 .GT. PERC1 .AND. DIF2 .GT. PERC2 ) GO TO 800
C
C      KFT = 2
C      NPT = KTAB(NTYPE)
C      A = 1115.
C      SUB = 0.0
C      XOUT = 0.0
C
C      IF ( DIF2 .LE. PERC2 ) SUB = SUBTR(NTYPE)
C
C      XIN = R*F*DV/(A*A)
C      IF(XIN.LT.X(2,NTYPE).OR.XIN.GT.X(NPT-1,NTYPE)) GO TO 700
C
C      D = 0.0
C      CALL TAIN(X(1,NTYPE), Y(1,NTYPE), XIN, XOUT, NPT, KFT, NER, D )

```

```

C
  700 CONTINUE
    DELDB = XOUT - SUB
C
    GO TO 900
  800 CONTINUE
    DELDB = 0.0
  900 CONTINUE
C
    RETURN
  END
*DECK OSPL
  SUBROUTINE OSPL (DECB,MIKES,NFREQ)
    REAL*8 DECB
    REAL*8 S,D,DDB
    DIMENSION DECB(70,35)
    DO 400 I1=1,MIKES
      S = DECB(I1,1)
      DO 390 L=2,NFREQ
        D=DECB(I1,L)
        IF (D.LT.40.) GO TO 390
        DDB=DABS(S-D)
        IF (DDB.GT.7.5) GO TO 380
        S=DEXP(1.1115)*DEXP(-.19077*DDB)+DMAX1(S,D)
        GO TO 390
      380 S=DEXP(1.1406)*DEXP(-.20172*DDB)+DMAX1(S,D)
      390 CONTINUE
        DECB(I1,34) = S
    400 CONTINUE
    RETURN
  END
*DECK OSPL1
  SUBROUTINE OSPL1 (DECB,MIKES,NFREQ)
    REAL*8 DECB
    REAL*8 S,D,DDB
    COMMON/IDENT/NP01,KMIC,NP01F,KLIG,IQLDEB,IQLFIN
    DIMENSION DECB(50,35)
    DO 400 I1=1,MIKES
      IF(IQLDEB.EQ.0) IQLDEB=1
      IF(IQLFIN.EQ.0) IQLFIN=NFREQ
      S = DECB(I1,IQLDEB)
      DO 390 L=IQLDEB+1,IQLFIN
        D=DECB(I1,L)
        IF (D.LT.40.) GO TO 390
        DDB=DABS(S-D)
        IF (DDB.GT.7.5) GO TO 380
        S=DEXP(1.1115)*DEXP(-.19077*DDB)+DMAX1(S,D)
        GO TO 390
      380 S=DEXP(1.1406)*DEXP(-.20172*DDB)+DMAX1(S,D)
      390 CONTINUE
        DECB(I1,34) = S
    400 CONTINUE
    RETURN
  END
*DECK OUTCOR
  SUBROUTINE OUTCOR(CORR,MIKEA,MIKEB,NFREQ)
C
  C    THIS SUBROUTINE PRINTS OUT THE ARRAY 'CORR' FOR NEAR AND FAR FIELDS .
  C
    REAL*8 CORR

```

```

      DIMENSION CORR(50,33)
C
C 400 FORMAT(1H1)
C
C      MIC = MIKEA
C
C      J=1 FOR NEAR-SIDELINE MIKES (MIKEA).
C      J=2 FOR FAR-SIDELINE MIKES (MIKEB).
C
C      IF ( MIC .GT. 13 ) MIC = 13
C      MICADD = MIC
C      MICNUM = MIC
C      M1 = 1
C
C      DO 2000 J = 1 , 2
C
C      IF ( J .EQ. 1 ) WRITE ( 6,430)
430  FORMAT (1H0,50X,23HNEAR FIELD MICROPHONES. /)
C      IF ( J .EQ. 2 ) WRITE ( 6,460)
460  FORMAT ( 1H1,50X,23H FAR FIELD MICROPHONES. / )
C
C      N1 = 1
C      MICNUM = MICADD
C
C      L=1 FOR THE 1ST 13 MIKES.
C      L=2 FOR THE REST OF THE MIKES.
C
C      DO 1500 L = 1 , 2
C
C      IF(L.EQ.1)WRITE(6,500) (1,1 = N1 , MICNUM )
500  FORMAT(5X,8HMIKE      , 8(11,8X),11,04(7X,12) )
C      IF(L.EQ.2)WRITE(6, 550) (1,1 = N1 , MICNUM )
550  FORMAT ( 5X,7HMIKE      , 12,11(7X,12) )
C      WRITE(6,600)
600  FORMAT(1X,4HBAND , / )
C
C      'K' REFERS TO THE MIKE NUMBER.
C      'I' REFERS TO THE BAND NUMBER.
C
C      DO 1000 I = 1 , NFREQ
C
C      WRITE(6,900) I , (CORR(K,I) , K = M1,MIC )
900  FORMAT(2X,12,4X,F7.2,12(2X,F7.2) )
1000 CONTINUE
C
C      IF ( MIKEA .EQ. MICADD ) GO TO 1700
C      M1 = MIC + 1
C      N1 = MICNUM + 1
C      MIC = MIKEA * J
C      MICNUM = MIKEA
C
C      WRITE(6,400)
C
C      1500 CONTINUE
C
C      1700 CONTINUE

```

```

C      N1 = 1
      M1 = MIKEA + 1
      MIC = MIKEA + MICADD
C
2000  CONTINUE
      WRITE(6,400)
C
      RETURN
      END
*DECK  OUTPUT
      SUBROUTINE OUTPUT (XDIS,DECB,ANGLES,FREQ,MIKES,NFREQ,HTMIKE)
C
      REAL*8 ANGLES,DECB,FREQ,HTMIKE,XDIS
      DIMENSION XDIS(70,2),DECB(70,35),ANGLES(70),FREQ(33),HTMIKE(70)
C
      LC = MIKES / 15
      IF(MOD(MIKES,15).NE.0) LC = LC + 1
      IST = 1
      DO 375 J = 1,LC
      ISTP = IST + 14
      IF (ISTP.GT.MIKES) ISTP=MIKES
C
      WRITE (6,610) ((K),K=IST,ISTP)
      WRITE (6,620) (ANGLES(K),K=IST,ISTP)
      WRITE (6,625) (HTMIKE(K),K=IST,ISTP)
      WRITE (6,630) (XDIS(K,1),K=IST,ISTP)
      WRITE (6,640) (XDIS(K,2),K=IST,ISTP)
      WRITE (6,650)
      DO 350 L=1,NFREQ
350  WRITE(6,670) FREQ(L), (DECB(K,L),K=IST,ISTP)
      WRITE(6,710) (DECB(K,34),K=IST,ISTP)
      WRITE(6,720) (DECB(K,35),K=IST,ISTP)
C
      WRITE (6,730)
      IST = ISTP + 1
375  CONTINUE
C
610  FORMAT (12H MICROPHONE:,14X,15(2X,12,2X))
620  FORMAT (12H ANGLE(DEG):,14X,15F6.1)
625  FORMAT (13H HEIGHT (FT):,13X,15F6.1)
630  FORMAT (14H CL  DIST(FT):,12X,15F6.1)
640  FORMAT (14H REF DIST(FT):,12X,15F6.1)
650  FORMAT (12H FREQ(HERTZ))
670  FORMAT (2X,F9.0,15X,15F6.1)
710  FORMAT (12H OVERALL SPL,14X,15F6.1)
720  FORMAT (5H PNDB,21X,15F6.1)
730  FORMAT (1H1)
      RETURN
      END
*DECK  POLFIT
      SUBROUTINE POLFIT(X,Y,M,N,U,C,IP)
C
C      -----
C
C      PURPOSE:  TO PERFORM A LEAST SQUARES POLYNOMIAL
C      CURVE FIT OF DEGREE N USING M GIVEN POINTS.
C
C      X      THE ARRAY CONTAINING THE X COORDINATES
C      Y      THE ARRAY CONTAINING THE Y COORDINATES

```

```

C      M   THE NUMBER OF POINTS TO BE FIT (LIMIT IS 100)
C      N   THE DEGREE OF FIT (LIMIT IS 15)
C      U   ON RETURN THE ARRAY OF N+1 COEFFICIENTS
C      C   ON RETURN THE ARRAY OF CORRECTED Y COORDINATES
C      IP  PRINT FLAG -- 1=ON (RESULTS ARE PRINTED), 0=OFF (NO OUTPUT)

```

```

THIS DECK IS INTENDED FOR USE ON THE CDC 7600. IF THIS PROGRAM
IS BEING USED ON TSS/360, REPLACE THE DIMENSION ARRAY1(13),
ARRAY2(12) CARD WITH :
DOUBLE PRECISION ARRAY1(13), ARRAY2(12)

```

```

-----
REAL*8 C,U,X,Y
REAL*8 Q,P,A,ALPH,B,S,G,ARRAY1,ARRAY2,D,XMEA,YMEA,HUH,XMEAN,YMEAN,
*ERR,YSTDER,E1,F1,W1,W,V,S1,T,T3,T5,ROOT,Q7,P0S,Q8,PCT
DIMENSION X(35),Y(35),Q(100),P(100),C(35)
DIMENSION A(16),ALPH(12),B(16),S(16),G(16),U(16)
DIMENSION ARRAY1(13),ARRAY2(12)
DATA ALPH/5H A ,5H + B ,5H + C ,5H + D ,5H + E ,5H + F ,
*5H + G ,5H + H ,5H + I ,5H + J ,5H + K ,5H + L /
DATA ARRAY1/5H ,5H X ,5H X**2,5H X**3,5H X**4,5H X**5,
*5H X**6,5H X**7,5H X**8,5H X**9,5H X**10,5H X**11,5H X**12/
DATA ARRAY2/5H FIRST,6H SECOND,5H THIRD,6H FOURTH,5H FIFTH,5H SIXTH,
*7H SEVENTH,6H EIGHTH,5H NINTH,5H TENTH,8H ELEVENTH,7H TWELFTH/
D=DFLOAT(M)
G(1)=0.0
N=N+1
IF (N.GT.12) GO TO 230
IF (M.LT.N) GO TO 240
XMEA=0.0
YMEA=0.0
HUH=0
DO 10 I=1,M
XMEA=XMEA+X(I)
YMEA=YMEA+Y(I)
10 HUH=HUH+Y(I)**2
XMEAN=XMEA/D
YMEAN=YMEA/D
ERR=(D*HUH-YMEA**2)/(D**2-D)
IF (IP.EQ.0) GO TO 500
YSTDER=DSQRT(ERR)
WRITE (6,250) M
WRITE (6,260) XMEAN,YMEAN
WRITE (6,270) YSTDER
500 DO 20 I=1,M
P(I)=0.0
20 Q(I)=1.0
DO 30 I=1,11
A(I)=0.0
B(I)=0.0
30 S(I)=0.0
E1=0.0
F1=0.0
W1=D
N4=12
I=1
40 W=0.0
DO 50 L=1,M
50 W=W+Y(L)*Q(L)

```

```

      S(I)=W/W1
      IF (I-N4.GE.0) GO TO 80
      IF (I-M.GE.0) GO TO 80
      E1=0.0
      DO 60 L=1,M
60    E1=E1+X(L)*Q(L)*Q(L)
      E1=E1/W1
      A(I+1)=E1
      W=0.0
      DO 70 L=1,M
      V=(X(L)-E1)*Q(L)-F1*P(L)
      P(L)=Q(L)
      Q(L)=V
70    W=W+V*V
      F1=W/W1
      B(I+2)=F1
      W1=W
      I=I+1
      GO TO 40
80    DO 90 L=3,12
90    G(L)=0.0
      G(2)=1.0
      LL=2
      DO 130 J=1,N
      S1=0.0
      DO 110 L=1,N
      IF (L.EQ.1) GO TO 100
      LL=L+1
      G(LL)=G(LL)-A(L)*G(LL-1)-B(L)*G(LL-2)
100   S1=S1+S(L)*G(LL)
110   CONTINUE
      U(J)=S1
      L=N+1
      DO 120 I2=2,N
      G(L)=G(L-1)
120   L=L-1
      LL=2
130   G(2)=0.0
      T=0.0
      DO 150 L=1,M
      C(L)=0.0
      J=N
      DO 140 I2=1,N
      C(L)=C(L)*X(L)+U(J)
140   J=J-1
      T3=Y(L)-C(L)
150   T=T+T3**2
      IF (M.NE.N) GO TO 160
      T5=0.0
      GO TO 170
160   T5=T/(D-DFLOAT(N))
      ROOT=DSQRT(T5)
      IF (IP.EQ.1) WRITE(6,280) ROOT
170   IF (DABS(ERR).LT.0.00001) ERR = 0.001
      Q7 = 1.0-T/(ERR*(D-1.0))
      LESS=N-1
      IF (IP.EQ.0) GO TO 501
      WRITE (6,290) LESS
      WRITE (6,300) Q7
      WRITE (6,310)

```



```

      DO 180 J=1,N
      I2=J-1
180  WRITE (6,320) I2,ALPH(J),U(J)
      N1 = N-1
      WRITE(6,330) ARRAY2(N1)
      N1 = N
      IF (N1.GT.9) N1=9
      WRITE(6,340) ALPH(1),ALPH(2),ARRAY1(2), (ALPH(I),ARRAY1(I),I=3,N1)
      IF (N.LE.9) GO TO 191
      WRITE(6,350) (ALPH(I),ARRAY1(I),I=10,N)
191  WRITE(6,360)
501  KOUNT = 0
      DO 220 L=1,M
      POS=DABS(Y(L)-C(L))
      IF (POS.LT.0.1E-08) Y(L)=C(L)
      Q8=Y(L)-C(L)
      IF (C(L).EQ.0.0) GO TO 200
      PCT=100.0*Q8/C(L)
      IF (IP.EQ.1) WRITE(6,370) X(L),Y(L),C(L),Q8,PCT
      GO TO 210
200  IF (IP.EQ.1) WRITE(6,380) X(L),Y(L),C(L),Q8
210  KOUNT=KOUNT+1
      IF (KOUNT.LT.42) GO TO 220
      IF (IP.EQ.1) WRITE(6,360)
      IF (KOUNT.EQ.42) KOUNT=0
220  CONTINUE
      N=N-1
      RETURN
230  WRITE (6,390)
      STOP
240  LESS=N-1
      WRITE (6,400) LESS
      RETURN
C
250  FORMAT (1H1,6X,18HNUMBER OF POINTS =,I4)
260  FORMAT (7X,17HMEAN VALUE OF X =,F10.4 / 7X,17HMEAN VALUE OF Y =,
*      F10.4)
270  FORMAT (7X,21HSTANDARD ERROR OF Y =,F10.4)
280  FORMAT (1H0,6X,35HSTANDARD ERROR OF ESTIMATE FOR Y =,F9.4 //)
290  FORMAT (7X,16HDEGREE OF FIT = ,I2)
300  FORMAT (7X,23HINDEX OF DETERMINATE = ,F9.7 /)
310  FORMAT (7X,4HTERM,4X,6HLETTER,10X,11HCOEFFICIENT/)
320  FORMAT (8X,12,6X,A3,7X,E23.16)
330  FORMAT (1H0,6X,12HEQUATION IS ,A8,18H DEGREE POLYNOMIAL /)
340  FORMAT (7X,3HY =,2A3,A1,1X,7(A3,A4,1X))
350  FORMAT (//1X,3(3H + ,A2,A5//))
360  FORMAT (1H1,9X,8HX-ACTUAL,7X,8HY-ACTUAL,8X,6HY-CALC,5X,
-      10HDIFFERENCE,5X,8HPCT-DIFF /)
370  FORMAT (7X,F12.6,3X,F12.6,3X,F12.6,3X,F10.6,5X,F8.4)
380  FORMAT (7X,F12.6,3X,F12.6,3X,F12.6,3X,F10.6,5X,8HINFINITE)
390  FORMAT (1X,38HERROR-----ELEVENTH DEGREE IS THE LIMIT)
400  FORMAT (1X,44HERROR-----TOO FEW POINTS FOR FITTING DEGREE,I3)
      END
*DECK PNDB
      SUBROUTINE PNDB(LP,NF,PLDB,DL,AL0,ANN,AN0)
C
C      THIS SUBROUTINE HAS BEEN MODIFIED TO WORK ONLY FOR ONE-THIRD
C      OCTAVE BAND WIDTH.
C
      REAL*8 AL0,ANN,AN0,DL,PLDB

```

```

      REAL*8 SUMN
      DIMENSION DL(24,2),ALD(24,2),ANN(24,2),AND(24,2)
      REAL*8 LP(27),LB(24),NOY(24),NMAX,NBAR
      N=NF
      IF(N.GT.24)N=24
      DO 21 I=1,24
21  LB(I)=LP(I)
      NMAX=0.0
      SUMN=0.0
      DO 13 I=1,N
      NOY(I)=DMIN1(AND(I,1)*ANN(I,1)**((LB(I)-ALD(I,1))/DL(I,1)),
1      AND(I,2)*ANN(I,2)**((LB(I)-ALD(I,2))/DL(I,2)))
      IF(NOY(I).GE.NMAX)NMAX=NOY(I)
13  SUMN=SUMN+NOY(I)
      NBAR = NMAX+0.15*(SUMN-NMAX)
      PLDB=40.0+10.0*DLOG10(NBAR)/DLOG10(2.0)
      IF(PLDB.LT.0.0)PLDB=0.0
      RETURN
      END
*DECK POLYX
      REAL*8 FUNCTION POLYX(COF,KFIT,X)
C
C      EVALUATE POLYNOMIAL OF ARBITRARY ORDER
C      USING HORNER'S RULE.
C
C      COF - COEFFICIENTS OF POLYNOMIAL
C      POLYX = COF(1) + COF(2)*X + COF(3)*X**2 + .....
C      (KFIT + 1 = NUMBER OF COEFFICIENTS.)
C
C      KFIT - HIGHEST POWER OF POLYNOMIAL
C
C      X - VALUE OF INDEPENDENT VARIABLE AT WHICH
C      POLYNOMIAL IS TO BE EVALUATED.C
      REAL*8 COF,X
      REAL*8 SUM
      DIMENSION COF(15)
C
      NP1=KFIT + 1
      SUM = COF(NP1)
      IF (KFIT.EQ.0) GOTO 20
C
      DO 10 I=1,KFIT
      J=NP1-I
      SUM=COF(J)+SUM*X
10  CONTINUE
C
20  POLYX = SUM
      RETURN
      END
*DECK READIN
      SUBROUTINE READIN(LIST,KPF,KODT,ITIT)
C
C      -----
C
C      SUBROUTINE READIN IS RESPONSIBLE FOR READING AND STORING INTO
C      COMMON THE DATA AND PARAMETERS TO BE USED FOR PROCESSING.
C      AS THE DATA IS READ IT IS ALSO PRINTED TO THE LINE PRINTER
C      RESULTS FILE (UNIT 06).
C
C      ENGINEERS INPUTS AND PARAMETERS ARE READ FROM LOGICAL UNIT

```

```

C      ASSIGNED TO VARIABLE LUN, USUALLY 5.
C
C      ANGLES AND DECIBEL READINGS FOR EACH MIKE ARE READ FROM
C      LOGICAL UNIT NUMBER IUN WHICH MAY OR MAY NOT BE THE SAME
C      AS LUN DEPENDING ON THE CONFIGURATION CHOSEN BY THE PROGRAMMER.
C
C      -----
C
      REAL*8 RFACT,V8,TTEMPF,SDTEMP,VTUN,RNZDIN,HTMIK,ALPHA,H,D
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
      *TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
      *COFNR,AN,YY,R2D,D2R,XDIS
      COMMON /SUB1/ VJ,          FREQ(33),THETAS(50,33),XOVERD(50,33),
      *          NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
      *          NZTYPE(10),NFREQ
C
      COMMON /SUB2/ PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
      1          FFSPL(50,35),ST(35),STP(35),
      3          IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
      4          SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
      5          ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
      6          PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
      7          R2D,D2R,          XDIS(70,2),IMRC,IBNC,IRC,IAAC,
      8          IPFRQC,IPANGL,ICALL,NZTYPE
      COMMON/TRAFLG/KTRAC,KTFFSPL,KTDECBM,KTDECBP
      COMMON/IDENT/NPOI,KMIC,NPOIF,KLIG,IGLDEB,IGLFIN
      DIMENSION ITIT(1)
C
      DATA IUN,LUN /5,5/
      DATA RFACT /459.6/
C
      WRITE(6,800)
      READ(IUN,910)NPOIF,KMICF,NPOI,KMIC,KLIG,LIST
      KPF=NPOIF
      910 FORMAT(2(15,1X,11),2(2)
      READ (IUN,901) NRUN,NZTYPE
      WRITE (6,801) NZTYPE,NTEST,NRUN,NZTYPE,NPOI,KMIC,NPOIF,KMICF
C
      READ (IUN,902) IPFRQC,IPANGL
      WRITE (6,802) IPFRQC,IPANGL
C
      READ (IUN,902) IMRC,IBNC,IRC,IAAC
      WRITE (6,807) IMRC,IBNC,IRC,IAAC
C
      LECTURE DES FLAG DE TRACE
      READ(IUN,902)KTRAC,KTFFSPL,KTDECBM,KTDECBP
      READ(IUN,910)(ITIT(I),I=1,20)
      WRITE(6,810)KTRAC,KTFFSPL,KTDECBM,KTDECBP
      READ (IUN,903) MIKEA,KFIT(1),KFIT(2),IWT1,IWT2,IGLDEB,IGLFIN
C
      READ (IUN,904) AN,YY,PSILO,PSIHI
      WRITE (6,804) AN,YY,PSILO,PSIHI
      AN=AN*3.2808
      YY=YY*3.2808
C
      READ (IUN,902) ITMPTP
C
      IF (ITMPTP.NE.0.AND.ITMPTP.NE.1) STOP 3
      IF (ITMPTP.NE.1) GOTO 200

```

```

      READ (IUN,905) V8,VAMB,THUM,TTEMPF,SDHUM,SDTEMP
C
C   THESE VARIABLES ARE SET EQUAL DUE TO NUMEROUS PROGRAM CHANGES.
C
      VJET = V8
      VTUN = VAMB
C
C
      TTEMP = TTEMPF+RFACT
      SDTEMP = SDTEMPF+RFACT
      GOTØ 210
200 READ (IUN,905) V8,VAMB,THUM,TTEMP,SDHUM,SDTEMP
      TTEMPF = TTEMP-RFACT
      SDTEMPF = SDTEMP-RFACT
210 RNZDIN = RNZDIA*12.0
C
      WRITE (6,806) V8,VTUN,THUM,SDHUM,RNZDIN,RNZDIA,TTEMPF,TTEMP,
*          SDTEMPF,SDTEMP
      READ (IUN,908) HTSRCE,HTMIK
      IF(KØDT.NE.Ø)GØ TØ 100
C   LECTURE LIGNE 1 -BRUIT DE FOND ET MESURE
      IF(KPF.EQ.Ø)GØ TØ 300
      KPT=1
      CALL RECHPOIN(KPT,NPOIF,KBID,NFREQ,MIKEA,KMICF,KLIG,DECB2,FREQ,
*ANGLES)
      KPT=2
300 CALL RECHPOIN(KPT,NPOI,KPF,NFREQ,MIKEA,KMIC,KLIG,DECB2,FREQ,
*ANGLES)
      GØ TØ 400
100 IF(KØDT.EQ.1)CALL LCART
      IF(KØDT.EQ.2)CALL RECHNASA(NFREQ,MIKEA,DECB2,FREQ,ANGLES)
400 WRITE(6,803)MIKEA,KFIT(1),KFIT(2),IWT1,IWT2
      DØ 101 J=1,MIKEA
101 HTMIKE(J)=HTMIK
      WRITE (6,809) HTSRCE,(HTMIKE(I),I=1,MIKEA)
809 FORMAT( / 1X,2ØHSOURCE HEIGHT (FT) =, F6.2 /
* 1X,2ØHMICROPHONE HEIGHT =, 15F6.2 / 2(21X,15F6.2/) 21X,15F6.2)
C
C
C
C   CALCULATION OF MIKE DISTANCES BASED ON
C   MIKE HEIGHT AND ANGLE ... SEE PROGRAM DOCUMENTATION
C   FOR REFERENCE ON METHOD USED (GEOMETRY)
C
      DØ 110 L=1,MIKEA
      IF (HTSRCE.EQ.HTMIKE(L)) GØTØ 112
      XDIS(L,2) = AN/DSIN((180.-ANGLES(L))*D2R)
      XDIS(L,1) = XDIS(L,2)*DCØS((180.-ANGLES(L))*D2R)
      GØTØ 110
112 IF (ANGLES(L).GT.90.) ALPHA=90.-ANGLES(L)
      IF (ANGLES(L).LT.90.) ALPHA=ANGLES(L)-90.
      H = HTSRCE-HTMIKE(L)
      IF (ANGLES(L).EQ.90.) GØTØ 113
      D = AN/DCØS(ALPHA*D2R)
      XDIS(L,1) = DSQRT(AN*AN+D*D)
C***** ECRITURE YY
      XDIS(L,2) = DSQRT(D*D+H*H)
      GØTØ 110
113 XDIS(L,1) = 0.0
      XDIS(L,2) = DSQRT(AN**2+H**2)

```

```

110 CONTINUE
    WRITE (6,805) (XDIS(L,1),L=1,MIKEA)
    WRITE (6,808) (XDIS(L,2),L=1,MIKEA)
C
C*****INPUT FORMATS
C
901 FORMAT(2I3)
902 FORMAT (6I1)
903 FORMAT (7I2)
904 FORMAT (4F6.2)
905 FORMAT (8F10.1)
908 FORMAT (16F5.0)
909 FORMAT (16F5.2)
1910 FORMAT(20A4)
C
C*****OUTPUT FORMATS
C
800 FORMAT(1H1,47HSOURCE LOCATION PROGRAM - AMBIENT VELOCITY CASE
* // 1X,
* 40HINPUT PARAMETERS READ FOR THIS EXECUTION /)
801 FORMAT(1X,16HIDENTIFICATION =,1X, 10A4,5X,4HTEST,14,5H RUN ,
* 13/1X,13HNOZZLE TYPE =,12,60X,6HPPOINT ,14,4H M ,11,5X,7HB DE F
*,14,4H M ,11)
802 FORMAT(1X,16HOUTPUT FLAGS =,1X,6I1)
803 FORMAT(/ 1X,19HNEAR FIELD MIKES = ,12 /
*1X,10HKFIT(1) = ,12,19X,10HKFIT(2) = ,12,/1X,
*10HIWT1 = ,12,19X,10HIWT2 = ,12)
804 FORMAT(// 12X,10HNEAR FIELD, 6X,9HFAR FIELD /
*1X,8HDISTANCE,F11.3,5X,F11.3,5X,7HPSILO =,F8.2,3X,7HPSIHI =,F8.2)
806 FORMAT(/ 1X,8HV JET =,F8.2,10X,7HV AMB =,F8.2 /
* 1X,8HTHUM =,F8.2,10X,7HSDHUM =,F8.2,10X,12HNOZZLE DIA =,
*F8.4,9H INCHES =,F8.4,5H FEET,
* // 21X,5HDEG F,10X,5HDEG R / 1X,11HTUNNEL TEMP,7X,
*F7.1,8X,F7.1 / 1X,12HSTD DAY TEMP,6X,F7.1,8X,F7.1)
805 FORMAT(// 1X,14HMIKE DISTANCES /
*1X,13HCENTER LINE =,10F10.3 / 3(14X,10F10.3 /))
807 FORMAT(1X,19HCORRECTION FLAGS =,1X,5I1)
808 FORMAT(/ 1X,13HSOURCE DIST =,10F10.3 / 3(14X,10F10.3 /))
810 FORMAT(' PLOT FLAGS = ',5I1)
END
*DECK LCART
SUBROUTINE LCART
REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
*TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
*COFNR,AN,YY,R2D,D2R,XDIS
COMMON /SUB1/ VJ, FREQ(33),THETAS(50,33),XOVERD(50,33),
* NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
* NZTYPE(10),NFREQ
C
COMMON /SUB2/ PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),
1 FFSPL(50,35),ST(35),STP(35),
3 IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
4 SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
5 ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
6 PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
7 R2D,D2R, XDIS(70,2),IMRC,IBNC,IRC,IAAC,
8 IPRQC,IPANGL,ICALL,NTYPE
C
IUN=LUN=5

```

```

      READ (IUN,903) NFREQ
903  FORMAT (7I2)
      READ (IUN,905) (FREQ(I),I=1,NFREQ)
905  FORMAT (8F10.1)
      DO 101 J=1,MIKEA
      READ (LUN,909) ANGLES(J), (DECB2(J,L),L=1,NFREQ)
909  FORMAT (16F5.2)
101  CONTINUE
      RETURN
      END
*    DECK RECHNASA
      SUBROUTINE RECHNASA(NFREQ,MIKEA,DECB,FREQ,ANGLES)
      COMMON/IDENT/IPT1,KBNSA
      COMMON/TITRE/LIB(10),IDATE(3),DB1(8,4)
      REAL*8 DECB(70,35),FREQ(33),ANGLES(70)
      DIMENSION ANG(30),SPE(30),IPAR(2),PAR(8),NZTYP(3),FQ(30)
      DIMENSION TAB1(13),TAB2(30),TAB3(31),TAB4(31),TAB5(31)
      EQUIVALENCE (TAB1(1),IPAR(1)),(TAB1(3),PAR(1)),(TAB1(11),NZTYP(1))
      EQUIVALENCE (TAB2(1),FQ(1))
      EQUIVALENCE (TAB3(1),ANG(1))
      EQUIVALENCE (TAB5(1),SPE(1))
      DATA ND1,ND2/2,3/,NDFQ/5/,LIB0/4H      /
      IF(KBNSA.EQ.1)NFREQ=25
      IF(KBNSA.EQ.2)NFREQ=23
C    LECTURE DU POINT CONTENANT LA LIGNE 1
      READ(ND1,101,END=601)(TAB1(I),I=1,13)
      IF(IPT1.NE.1)PAR(1)WRITE(6,104)IPAR(1);STOP
104  FORMAT(' ERREUR POINT LU =',I5)
      NP01=IPAR(1)
      N=IPAR(2)
      MIKEA=N
101  FORMAT(31A4)
      DB1(1,1)=0.
      DB1(1,2)=PAR(5)/3.2808
      READ(ND1,101)(TAB2(I),I=1,30)
      DO 10 I=1,NFREQ
10   FREQ(I)=FQ(I+NDFQ)
      READ(ND1,101)(TAB3(I),I=1,N)
      DO 20 I=1,N
20   ANGLES(I)=ANG(I)
      READ(ND1,101)(TAB4(I),I=1,N)
      DO 30 IA=1,N
      READ(ND1,101)(TAB5(I),I=1,31)
      DO 40 IDB=1,NFREQ
40   DECB(IA,IDB)=SPE(IDB+NDFQ)
30   CONTINUE
      DO 5 I=1,3
5    LIB(I)=NZTYP(I)
      DO 6 I=4,10
6    LIB(I)=LIB0
      DO 7 I=1,3
7    IDATE(I)=0
      GO TO 900
601  WRITE(6,102)ND1
102  FORMAT(' FIN DE FICHIER ETIQ. LOGIQUE ',I2)
      STOP
900  RETURN
      END
*DECK RECHP01N
      SUBROUTINE RECHP01N(KPT,NP01,KF,NFREQ,MIKEA,KMIC,KLIG,DECB,FREQ,

```

```

      *ANGLES)
C
C      INTERFACE PROG AMES ET PROG ONERA
C      PAS DE CALIBRATION SUR LES MESURES BRUTES
C      RETRAIT DU BRUIT DE FOND SUIVANT CODAGE
C      LECTURE BRUIT DE FOND AVANT MESURE
C
C      NO MICROS
C      IPI(1)=0- 7M,LIGNE 1,MICRO 1      1
C      IPI(2)=0- 7M,LIGNE 1,MICRO 2      2
C      IPI(3)=0- 7M,LIGNE 2,MICRO 1      3
C      IPI(4)=0- 7M,LIGNE 2,MICRO 2      4
C      IPI(5)=7-14M,LIGNE 1,MICRO 1      5
C      IPI(6)=7-14M,LIGNE 1,MICRO 2      6
C      IPI(7)=7-14M,LIGNE 2,MICRO 1      7
C      IPI(8)=7-14M,LIGNE 2,MICRO 2      8
C
      REAL*8 OMEGA(8,40),DECB(70,35),FREQ(33),ANGLES(70)
      REAL*8 AUX(70,30),DECBF(70,30),TM,TF
      COMMON/TST/ITEST(70,30),NOFR(30)
      COMMON/TITRE/LIB(10),IDATE(3),DB1(8,4),IPI(8)
      DIMENSION LTT(10),JFR(44),IRG(2)
      DIMENSION IDB1(8,6),TETA(8,40),SDB(8,40,50)
      DIMENSION TAB1(050),TAB2(050),ITAB(6),TAB(4),ANG(40),SPE(45)
      EQUIVALENCE(TAB1(1),TITR)
      EQUIVALENCE(TAB1(2),ITAB(1))
      EQUIVALENCE(TAB1(8),TAB(1))
      EQUIVALENCE(TAB1(12),LTT(1))
      EQUIVALENCE(TAB1(22),ANG(1))
      EQUIVALENCE(TAB2(1),ANGLE)
      EQUIVALENCE(TAB2(2),SPE(1))
C
      DATA IPI/1,2,3,4,5,6,7,8/,NBPIS/4/
C
      DATA JFR/3,4,5,6,8,10,12,16,20,25,31,40,50,63,80,100,125,160,200,
      *250,315,400,500,630,800,1000,1250,1600,2000,2500,3150,4000,5000,
      *6300,8000,10000,12500,16000,20000,25000,31500,40000,50000,63000/
      DATA IUN/5/
C
      FREQUENCE 80 A 20000HZ
C
      ND=2
      NUMER =1
      REWIND ND
50  KODR=0
600 CONTINUE
      READ(ND,105,END=610)(TAB1(I),I=1,50)
105  FORMAT(50A4)
      IF(ITAB(1).NE.NPO1 )GO TO 500
C      NOUVELLE INDEXATION
      INDEX=ITAB(4)
C      MISE EN TABLEAU DU TITRE
      DO 12 I=1,10
12  LIB(I)=LTT(I)
      IDJ=ITAB(2)/4096
      IDF=ITAB(2)-IDJ*4096
      IUJ=IDF/256
      IDF=IDF-IUJ*256
      IDATE(1)=IDJ*10+IUJ
      IDM=IDF/16
      IUM=IDF-IDM*16

```

```

        IDATE(2)=IDM*10+IUM
        IDATE(3)=79
        DO 20 I=1,6
20      IDB1(INDEX,I)=ITAB(I)
        DO 22 I=1,4
22      DB1(INDEX,I)=TAB(I)
        DO 24 I=1,ITAB(5)
24      TETA(INDEX,I)=ANG(I)
C      REMISE EN ORDRE DES SPECTRESB SI BANDE EN SENS INVERSE
        DO 250 LA=1,ITAB(5)
        IF(ITAB(6).GT.1) IA=ITAB(5)+1-LA;GO TO 170
        IA=LA
C      LECTURE DES SPECTRES
170 CONTINUE
        READ(ND,105,END=630)(TAB2(I),I=1,50)
C      RANGEMENT DES ANGLES ET DES SPECTRES
        OMEGA(INDEX,IA)=ANGLE
        DO 250 IDB=1,45
250      SDB(INDEX,IA,IDB)=SPE(IDB)
        NUMER=NUMER+1
        IF(NUMER.LE.NBP1ST) GO TO 50
        GO TO 700
500      DO 25 J=1,ITAB(5)
        READ(ND,105,END=620)(TAB2(I),I=1,50)
        25 CONTINUE
        GO TO 600
610      KDDR=1
        GO TO 650
620      KDDR=2
        GO TO 650
630      KDDR=3
650      WRITE(6,104)KDDR;STOP
104      FORMAT(' ERREUR EN LECTURE BANDE KDDR=',I5)
700 CONTINUE
C      MISE EN TABLEAU DES FREQ DES ANGLES ET SPECTRES DE 0A14 METRES
C      RANG-1 DE LA PREMIERE FREQ TRAITEE
        NFDEC=14
        NFREQ=25
        DO 5 IDB=1,NFREQ
5      FREQ(IDB)= JFR(IDB+NFDEC)
C      CHOIX DU TRAITEMENT
C      KMIC=0 MICROS A+ B MOYENNES
C      KMIC=1 MICRO A
C      KMIC=2 MICRO B
C
C
        IA=IB=0
C      LIGNE 1 OU 2
        IF(KMIC.NE.0.AND.KMIC.NE.1)GO TO 1000
C      MICRO A
        IF(KLIG.EQ.1) M1=IPI(1);M2=IPI(5)
        IF(KLIG.EQ.2) M1=IPI(3);M2=IPI(7)
        NT1=IDB1(M1,5)
        I=1
        LEGAL=0
        DO 10 K=1,IDB1(M1,5)
        K1=INT(OMEGA(M1,K)+0.5)
        K2=INT(OMEGA(M2,I)+0.5)
        IF(K1.EQ.K2) I=I+1;LEGAL=LEGAL+1
10 CONTINUE

```



```

C   SECTEUR 0-7M
    D0 30 I=1, IDB1(M1,5)
    IA=IA+1
    ANGLES(IA)=OMEGA(M1,I)
    D0 30 IDB=1,NFREQ
30  DECB (IA,IDB)=SDB(M1,I ,IDB+NFDEC)
C   SECTEUR 7-14M
    D0 40 J=LEGAL +1, IDB1(M2,5)
    IA=IA+1
    ANGLES(IA)=OMEGA(M2,J)
    D0 40 IDB=1,NFREQ
40  DECB (IA,IDB)=SDB(M2,J,IDB+NFDEC)
    IF(KMIC.NE.0)G0 T0 2000
C   MICRO B
1000 IF(KLIG.EQ.1) M1=IPI(2);M2=IPI(6)
    IF(KLIG.EQ.2) M1=IPI(4);M2=IPI(8)
    NT1=IDB1(M1,5)
    I=1
    LEGAL=0
    D0 11 K=1, IDB1(M1,5)
    K1=INT(OMEGA(M1,K)+0.5)
    K2=INT(OMEGA(M2,I)+0.5)
    IF(K1.EQ.K2) I=I+1;LEGAL=LEGAL+1
11  CONTINUE
C   SECTEUR 0-7M
    D0 31 I=1, IDB1(M1,5)
    IB=IB+1
    ANGLES(IB)=OMEGA(M1,I)
    D0 31 IDB=1,NFREQ
31  AUX(IB,IDB)=SDB(M1,I,IDB+NFDEC)
C   SECTEUR 7-14M
    D0 41 J=LEGAL+1, IDB1(M2,5)
    IB=IB+1
    ANGLES(IB)=OMEGA(M2,J)
    D0 41 IDB=1,NFREQ
41  AUX(IB,IDB)=SDB(M2,J,IDB+NFDEC)
    IF(KMIC.NE.0)G0 T0 3000
    MIKEA=IA
C   MOYENNE MICRO A + MICRO B
    D0 70 I=1,MIKEA
    D0 70 IDB=1,NFREQ
70  DECB(I,IDB)=(DECB(I,IDB)+AUX(I ,IDB))/2.
    G0 T0 5000
2000 MIKEA=IA
    G0 T0 5000
3000 MIKEA=IB
    D0 72 I=1,MIKEA
    D0 72 IDB=1,NFREQ
72  DECB(I,IDB)=AUX(I,IDB)
5000 IF(MIKEA.GT.70)WRITE(6,106)MIKEA;STOP
106 FORMAT(' TROP D ANGLES LIGNE 1 OU 2 =',I5)
    IF(KF.EQ.0)RETURN
C   SAUVEGARDE BRUIT DE FOND
    IF(KPT.NE.1)G0 T0 900
    D0 80 I=1,MIKEA
    D0 80 IDB=1,NFREQ
80  DECBF(I,IDB)=DECB(I,IDB)
    RETURN
900 CONTINUE
    D0 81 I=1,MIKEA

```

```

      DO 81 IDB=1,NFREQ
81  ITEST(1,IDB)=0
C    RETRAIT BRUIT DE FOND
      DO 82 I=1,MIKEA
      DO 82 IDB=1,NFREQ
      DELT=DECB(1,IDB)-DECBF(1,IDB)
      IF(DELT.LE.0.)DECB(1,IDB)=0.;GO TO 82
      IF(DELT.LT.5.)ITEST(1,IDB)=1000
      TM=DECB(1,IDB)
      TF=DECBF(1,IDB)
      DECB(1,IDB)=10.*DLOG10(10.*(TM/10.)-10.*(TF/10.))
82  CONTINUE
      RETURN
      END
*DECK STEST
      SUBROUTINE STEST(MIKES,NFREQ,ANGLES,FREQ)
C
      REAL*8 ANGLES(70),FREQ(33)
      COMMON/TST/ITEST(70,30)
C    TEST BRUIT DE FOND
      WRITE(6,605)
605  FORMAT(1H1// ' TEST BRUIT DE FOND'//)
      LC = MIKES / 15
      IF(MOD(MIKES,15).NE.0) LC = LC + 1
      IST = 1
      DO 375 J = 1,LC
      ISTP = IST + 14
      IF (ISTP.GT.MIKES) ISTP=MIKES
C
      WRITE (6,610) ((K),K=IST,ISTP)
      WRITE (6,620) (ANGLES(K),K=IST,ISTP)
      WRITE (6,650)
      DO 350 L=1,NFREQ
350  WRITE(6,670) FREQ(L), (ITEST(K,L),K=IST,ISTP)
C
      WRITE (6,730)
      IST = ISTP + 1
375  CONTINUE
C
610  FORMAT (12H MICROPHONE: ,14X,15(2X,12,2X))
620  FORMAT (12H ANGLE(DEG): ,14X,15F6.1)
650  FORMAT (12H FREQ(HERTZ))
670  FORMAT (2X,F9.0,15X,15(2X,12,2X))
730  FORMAT (1H1)
      RETURN
      END
*DECK SPCTRA
      SUBROUTINE SPCTRA
C
      REAL*8 PRESS,FTOM,CONST,RTOK,DEGK,DV,SLD,ABDBM,DELDDB,TERM,SPL,
      *PHI2,DIST,DISTN,ABSCOR
      REAL*8 VJ,FREQ,THETAS,XOVERD,XDMAX,THETSM,RNZDIA
      REAL*8 PSIS,XDN,PSI1,FFSPL,ST,STP,PSI2,VAMB,VJET,SDHUM,SDTEMP,
      *TTEMP,THUM,PSILO,PSIHI,ANGLES,DECB2,HTSRCE,HTMIKE,PSIVAL,XDVAL,
      *COFNR,AN,YY,R2D,D2R,XDIS
      COMMON /SUB1/ VJ, FREQ(33),THETAS(50,33),XOVERD(50,33),
      * NTAB(33),NTEST,XDMAX(33),THETSM(33),RNZDIA,
      * NZTYPE(10),NFREQ
C
      COMMON /SUB2/ PSIS(50,35),XDN(50,35),PSI1(50,35),PSI2(50,35),

```

```

1          FFSPL(50,35),ST(35),STP(35),
3          IWT1,IWT2,VAMB,VJET,KFIT(2),SDHUM,
4          SDTEMP,TTEMP,THUM,PSILO,PSIHI,MIKEA,IBIDON,
5          ANGLES(70),DECB2(70,35),HTSRCE,HTMIKE(70),
6          PSIVAL(50),XDVAL(50,35),COFNR(11,35),AN,YY,
7          R2D,D2R,          XDIS(70,2),IMRC,IBNC,IRC,IAAC,
8          IPFRQC,IPANGL,ICALL,NTYPE

C
C
      PRESS = 1.0
      FCOM = 0.3048
      CONST = 20.0*DLGG10(YY/AN)
      RTOK = 0.55555556
      DEGK = TTEMP*RTOK
      DV = VJET - VAMB
      SLD = AN / RNZDIA

C
C
      PASS THROUGH EACH FREQUENCY AND DEFINE THE FAR FIELD SPECTRA
C
      IFCNT = 50
      DO 200 J=1,NFREQ
C
      IF (NTAB(J).EQ.0) GOTO 250
      WRITE(6,600) FREQ(J),ST(J)
600  FORMAT(1H1,12H FREQUENCY =,F9.0,18X,14HSTROUHAL NO. =,F9.3,17X,
      *      26HRADIAL DISTANCE CORRECTION )
      WRITE(6,650) STP(J),CONST
650  FORMAT(40X,14HST-PRIME NO. =,F9.3,17X,15H20*LOG(YY/AN) =,F9.4/)
      WRITE(6,675)
675  FORMAT(109X, 9HCORRECTED )
      WRITE(6,700)
700  FORMAT(58X,14HNEAR FIELD SPL,5X,10HABSORPTION,5X,10HNEAR FIELD,
      *      5X,13HFAR FIELD SPL )
      WRITE(6,725)
725  FORMAT(5X,3HNO.,6X,5HPSI-S,8X,2HX2,8X,4HPSI1,8X,4HPSI2,8X,
      *      8HAT PSI-S,8X,2(10HCORRECTION,5X),3X,8HAT PSI-S )

C
C
      FIND PSI1 (PSI-11) FOR EACH PSIVAL, XDVAL
      PAIR KNOWN FOR THIS FREQUENCY
C
      DO 100 IROW = 1,IFCNT
C
      ABDBM = 0.0
      DELDB = 0.0
C
      IF ( PSIVAL(IROW) .NE. 90. ) GO TO 140
      PSI1(IROW,J) = 90.
      PSI2(IROW,J) = 90.
      GO TO 175
C
140  CONTINUE
      IF ( PSIVAL(IROW) .EQ. 180. ) GO TO 145
      TERM = DTAN((180.-PSIVAL(IROW))*D2R)
      TERM = (AN/TERM)+XDVAL(IROW,J)
      IF (TERM.NE.0.) GOTO 150
145  CONTINUE
      PSI1(IROW,J) = 0.0
      GOTO 161

```

```

150 PSI1(IROW,J) = -1.0*(DATAN(AN/TERM))*R2D
    IF ( PSI1(IROW,J) .LT. 0. ) PSI1(IROW,J) = PSI1(IROW,J) + 180.
C
C
C   DETERMINE PSI-12 FOR EACH KNOWN PSI-11
C   PSI-12 WILL BE USED TO COMPUTE ARBITRARY FAR FIELD SPL
C
    IF ( PSIVAL(IROW) .EQ. 180. ) GO TO 161
    TERM = DTAN((180.-PSIVAL(IROW))*D2R)
    TERM = (YY/TERM)+XDVAL(IROW,J)
    IF (TERM.EQ.0.) GO TO 161
    PSI2(IROW,J) = -1.0*(DATAN(YY/TERM))*R2D
    IF ( PSI2(IROW,J) .LT. 0. ) PSI2(IROW,J) = PSI2(IROW,J) + 180.
    GO TO 175
161 PSI2(IROW,J) = 0.0
    GO TO 171
C
C
C   EVALUATE NEAR FIELD SPL AND ADJUST
C   FOR DISTANCE FOR PSI2 JUST COMPUTED
C
175 SPL = POLYX(COFNR(1,J),KFIT(2),PSIVAL(IROW))
    FFSPL(IROW,J) = SPL-CONST
C
C   ADJUST FOR ATMOSPHERIC ATTENUATION
C
    PHI2 = 180. - PSIVAL(IROW)
C
C
    DIST = YY/DSIN(PHI2*D2R)
    CALL AIFAB(PRESS,DEGK,SDHUM,FREQ(J),ABDBM)
    FFSPL(IROW,J) = FFSPL(IROW,J)-ABDBM*DIST*FTOM
    DISTN = AN/DSIN(PHI2*D2R)
    CALL NFCORR(NTYPE,SLD,DISTN,FREQ(J),DV,DELDDB )
    FFSPL(IROW,J) = FFSPL(IROW,J) - DELDDB
C
    GO TO 170
171 FFSPL(IROW,J) = 0.0
C
170 CONTINUE
    ABSCOR = ABDBM*FTOM*DIST
C
    WRITE(6,802)IROW,PSIVAL(IROW),XDVAL(IROW,J),PSI1(IROW,J),
      * PSI2(IROW,J),SPL,ABSCOR,DELDDB,FFSPL(IROW,J)
802 FORMAT(5X,12,5X,F7.2,5X,F6.2,2(5X,F7.2) ,8X,2(F7.2, 9X),
      * F6.2,10X,F7.2 )
C
100 CONTINUE
    CALL WRTEQN (KFIT(2),COFNR(1,J))
    GO TO 200
250 DO 110 IROW=1,IFCNT
110 FFSPL(IROW,J)=0.
200 CONTINUE
C
    RETURN
    END
*DECK SUBPDB
SUBROUTINE SUBPDB (MIKES,DECIBL,NFREQ)
REAL*8 DECIBL
REAL*8 DL,ALO,ANN,ANO,PDB

```

```

REAL*8 LP
DIMENSION DECIBL(70,35),LP(27)
DIMENSION DL(24,2),ALG(24,2),ANN(24,2),ANO(24,2)
DATA DL/15*10.,9*110.,30.,25.,2*26.
1      ,28.,2*27.,30.,51.,6*10.,7*110.,6.,9./
DATA ALG/52.,51.,49.,47.,46.,45.,43.,42.,41.,5*40.,38.,34.
1      ,32.,30.,2*29.,30.,31.,34.,37.,64.,60.,56.,53.,51.,48.
2      ,46.,44.,42.,5*40.,38.,34.,32.,30.,2*29.,30.,31.,37.,41./
DATA ANN/15*2.,9*1975.,13.5,10.3,2*9.07,9.76,2*7.94,9.15,
1      136.7,6*2.,7*1975.,1.79,2.4/
DATA ANO/48*1.0/

C
L24 = 24
IF ( NFREQ .LT. 24 ) L24 = NFREQ
DO 280 II = 1,MIKES
KSW=0
DO 260 L=1,L24
LP(L) = DECIBL(II,L+6)
IF (LP(L).GT.0.0) KSW = 1
IF (LP(L).LT.0.0) LP(L) = 0.0
260 CONTINUE
IF (KSW.GT.0) GO TO 270
PDB=0.
GO TO 280
270 CONTINUE
NF = L24
CALL PNDB(LP,NF,PDB,DL,ALG,ANN,ANO)
280 DECIBL(II,35) = PDB
RETURN
END
*DECK TAINI
SUBROUTINE TAINI (XTAB, FTAB, X, FX, N, K, NER, MON)
REAL*8 FTAB,FX,X,XTAB
REAL*8 T,C
DIMENSION XTAB(1), FTAB(1), T(10), C(10)
REAL*8 MON
IF (N-K) 1, 1, 2
1 NER = 2
RETURN
2 IF (K-9) 3, 3, 1
3 IF (MON) 4, 4, 5
5 IF (MON-2.) 6, 7, 4
4 J = 0
NM1 = N-1
DO 8 I = 1, NM1
IF (XTAB(I) - XTAB(I+1)) 9, 11, 10
11 NER = 3
RETURN
9 J = J - 1
GO TO 8
10 J = J + 1
8 CONTINUE
MON = 1.
IF (J) 12, 6, 6
12 MON = 2.
7 DO 13 I = 1, N
IF (X-XTAB(I)) 14, 14, 13
14 J = 1
GO TO 18
13 CONTINUE

```

```

      GO TO 15
6 DO 16 I = 1, N
  IF (X-XTAB(I)) 17, 17, 16
17 J = I
  GO TO 18
16 CONTINUE
15 J = N
18 J = J-(K+1)/2
  IF (J) 19, 19, 20
19 J = 1
20 M = J+K
  IF (M-N) 21, 21, 22
22 J = J - 1
  GO TO 20
21 KP1 = K + 1
  JSAVE = J
  DO 23 L = 1, KP1
    C(L) = X-XTAB(J)
    T(L) = FTAB(J)
23 J = J + 1
  DO 24 J = 1, K
    I = J + 1
25 T(I) = (C(J) * T(I) - C(I) * T(J)) / (C(J) - C(I))
    I = I + 1
  IF (I - KP1) 25, 25, 24
24 CONTINUE
  FX = T(KP1)
  NER = 1
  RETURN
  END
*DECK THMERG
SUBROUTINE THMERG
C
C   INSERT THETSM INTO THETAS AND INSERT XDMAX INTO XOVERD
C
C
  REAL*8 VJ, FREQ, THETAS, XOVERD, XDMAX, THETSM, RNZDIA
  REAL*8 PSIS, XDN, PS11, FFSP, ST, STP, PS12, VAMB, VJET, SDHUM, SDTEMP,
  *TTEMP, THUM, PS10, PS1H, ANGLES, DECB2, HTSRCE, HTMIKE, PSIVAL, XDVAL,
  *COFNR, AN, YY, R2D, D2R, XDIS
  COMMON /SUB1/ VJ, FREQ(33), THETAS(50,33), XOVERD(50,33),
  * NTAB(33), NTEST, XDMAX(33), THETSM(33), RNZDIA,
  * NZTYPE(10), NFREQ
C
  COMMON /SUB2/ PSIS(50,35), XDN(50,35), PS11(50,35), PS12(50,35),
  1 FFSP(50,35), ST(35), STP(35),
  3 IWT1, IWT2, VAMB, VJET, KFIT(2), SDHUM,
  4 SDTEMP, TTEMP, THUM, PS10, PS1H, MIKEA, IBIDON,
  5 ANGLES(70), DECB2(70,35), HTSRCE, HTMIKE(70),
  6 PSIVAL(50), XDVAL(50,35), COFNR(11,35), AN, YY,
  7 R2D, D2R, XDIS(70,2), IMRC, IBNC, IRC, IAAC,
  8 IPFRQC, IPANGL, ICALL, NTYPE
C
  DO 500 J = 1, NFREQ
C
  NTB = NTAB(J)
  IF ( NTB .GT. 50 ) NTB = 50
  NTBADI = NTB + 1
C
  DO 400 I = 1, NTB

```

```

C      IF ( THETSM(J) .EQ. THETAS(I,J) ) GO TO 500
C      IF ( THETSM(J) .GT. THETAS(I,J) ) GO TO 400
C      IL = NTBAD1 - 1
C
C      MOVE THE REST OF THE ELEMENTS FORWARD SO THAT THE APPROPRIATE PLACE IS
C      FREED FOR THE NEW VALUE.
C      IF THE ARRAY HAS 50 ELEMENTS ALREADY , THE LAST ELEMENT WILL BE LOST TO
C      MAKE ROOM FOR THE NEW VALUE.
C
C      DO 300 L = 1 , IL
C      LTB = NTBAD1 - L
C      IF ( LTB .EQ. 50 ) GO TO 300
C      THETAS(LTB+1,J) = THETAS(LTB,J)
C      XOVERD(LTB+1,J) = XOVERD(LTB,J)
300  CONTINUE
C
C      INSERT THE SINGLE VALUE INTO THE EMPTIED SLOT
C
C      THETAS(I,J) = THETSM(J)
C      XOVERD(I,J) = XDMAX(J)
C      NTAB(J) = NTAB(J) + 1
C      GO TO 500
C
C      400 CONTINUE
C
C      CHECK IF THE SINGLE VALUE IS GREATER THAN THE LAST ELEMENT OF THE ARRAY.
C
C      IF ( THETSM(J) .LE. THETAS(NTB,J) ) GO TO 500
C      LST = NTB+1
C      IF(NTB .EQ. 50 ) LST = 50
C      THETAS(LST,J) = THETSM(J)
C      XOVERD(LST,J) = XDMAX(J)
C      NTAB(J) = NTAB(J) + 1
C
C      500 CONTINUE
C      RETURN
C      END
C      *DECK WRTEQN
C      SUBROUTINE WRTEQN (N,COEF)
C
C      REAL*8 COEF
C      REAL*8 ALPH,ARRAY1,ARRAY2
C      DIMENSION COEF(12),ALPH(12),ARRAY1(13),ARRAY2(12)
C
C      DATA ALPH/5H A ,5H+ B ,5H+ C ,5H+ D ,5H+ E ,5H+ F ,5H+ G
C      *,5H+ H ,5H+ I ,5H+ J ,5H+ K ,5H+ L /
C      DATA ARRAY1/5H ,5HX ,5HX**2 ,5HX**3 ,5HX**4 ,5HX**5 ,5HX**6
C      *,5HX**7 ,5HX**8 ,5HX**9 ,5HX**10 ,5HX**11 ,5HX**12/
C      DATA ARRAY2/5H FIRST,6HSECOND,5HTHIRD,6HFOURTH,5HFIFTH,5HSIXTH,
C      17HSEVENTH,6HEIGHTH,5HNINTH,5HTENTH,8HELEVENTH,7HTWELFTH/
C      NN = N + 1
C
C      WRITE (6,101)
101  FORMAT(// 1X,32HEQUATION DEFINING NEAR FIELD SPL /
C      * 1X,27H(WHERE Y=SPL, X=ANGLE PSIS) /)

```

```

        WRITE (6,310)
        DO 180 J=1,NN
            I2=J-1
180    WRITE (6,320) I2,ALPH(J),COEF(J)
        N1 = NN-1
        WRITE(6,330) ARRAY2(N1)
        N1 = NN
        IF (N1.GT.9) N1=9
C
        WRITE(6,340) ALPH(1),ALPH(2),ARRAY1(2),(ALPH(1),ARRAY1(1),I=3,N1)
        IF (NN.LE.9) GOTO 191
        WRITE(6,350) (ALPH(1),ARRAY1(1),I=10,NN)
C
191    RETURN
310    FORMAT (7X,4HTERM,4X,6HLETTER,10X,11HCoefficient/)
320    FORMAT (8X,12,6X,A3,7X,E23.16)
330    FORMAT (1H0,6X,12HEQUATION IS ,A8,18H DEGREE POLYNOMIAL /)
340    FORMAT (7X,3HY =,2A3,A1,1X,7(A3,A4,1X))
350    FORMAT (//1X,3(3H + ,A2,A5//))
    END
*DECK TDECB1
    SUBROUTINE TDECB1(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,
    1ITIT,NCAR,XMOD)
C*****
C***** SOUS-PROGRAMME DE TRACE DES COURBES DECB=F(ANGLES)
C*****
    REAL*8 DECB(70,35),ANGLES(70),FREQ(33)
    COMMON /IDENT/NP01,KMIC,NP01F,KLIO
    COMMON /TITRE/LIB(10),IDATE(3),DB1(8,4),IPI(8)
    DIMENSION ITIT(1),XMOD(1)
C*****
    DIMENSION SANG(130)
    DIMENSION NAM(7),NASA(5),IPOINT(2),IX(2),IY(2),NMIC(2),IDEG(1),ITH
    *ETAS(2)
    DIMENSION MICRO(4),MOINS(1)
    DIMENSION IANG(4),XIANG(8),ISDB(1),IFREQ(2),IA(1),IHZ(1)
    DIMENSION X12(2),Y12(2)
    DIMENSION DIFF(70)
    DIMENSION NTR(15),NEP(15),NPL(15)
C*****
    DATA CX/300./,CY/210./
    DATA NAM/'P9-J.COURATIN  PROG.NOISE4 '/
    DATA NASA/'N.A.S.A./O.N.E.R.A. '/
    DATA IPOINT/'POINT  '/,IX/'X=  M'/,IY/'Y=  M'/,NMIC/' MICRO
    *= '/'
    DATA IANG/50,90,130,170/
    DATA XIANG/13.,43.,73.,103.,153.,183.,213.,243./
    DATA IDEG/'DEG '/,ISDB/'5DB '/,IFREQ/'FREQ.  '/,ITHETAS/'THETAS
    *'/
    DATA MOINS/'-  '/,IHZ/'HZ  '/,IA/'A  '/
    DATA NTR/1,2,3,4,5,1,2,3,4,5,1,2,3,4,5/
    DATA NEP/1,2,1,1,1,1,2,1,1,1,1,2,1,1,1/
    DATA NPL/1,1,1,1,1,2,2,2,2,3,3,3,3,3/
C***** TRAITEMENT LIGNE 1
    ILIGN=1
C***** INITIALISATION DES VALEURS X , Y , SANG ET MICRO
C***** INITIALISATION DES VALEURS POUR LA LIGNE 1
140    CONTINUE
        IF(KLIO.EQ.1)X=DB1(IPI(1),1);Y=DB1(IPI(1),2)
        IF(KLIO.EQ.2)X=DB1(IPI(3),1);Y=DB1(IPI(3),2)

```



```

      DO 150 I=1,MIKEA
      SANG(I)=SNGI(ANGLES(I))
      DIFF(I)=SNGI(DECBI,34))
150  CONTINUE
      NBPT=MIKEA
      IF(KLIB.EQ.2) GO TO 260
      IF(KMIC.EQ.1) MICRO(I)=IPI(1) ; MICRO(2)=IPI(5) ; GO TO 160
      IF(KMIC.EQ.2) MICRO(I)=IPI(2) ; MICRO(2)=IPI(6) ; GO TO 160
      MICRO(1)=IPI(1)
      MICRO(2)=IPI(5)
      MICRO(3)=IPI(2)
      MICRO(4)=IPI(6)
      GO TO 160
260  IF(KMIC.EQ.1) MICRO(1)=IPI(3) ; MICRO(2)=IPI(7) ; GO TO 160
      IF(KMIC.EQ.2) MICRO(1)=IPI(4) ; MICRO(2)=IPI(8) ; GO TO 160
      MICRO(1)=IPI(3)
      MICRO(2)=IPI(7)
      MICRO(3)=IPI(4)
      MICRO(4)=IPI(8)
C***** NOMBRE DE FREQUENCES PAR LIGNE
160  CONTINUE
      NU=1
      CALL MINMAX(DIFF,NBPT,NU,DMIN,DMAX,XMOD)
      IDMAX=(DMAX/10.)+1
      IDMAX=IDMAX*10
      IDMIN=IDMAX-70
      DMAX=IDMAX
      DMIN=IDMIN
      IF(NFREQ.GT.25) GO TO 170
      IFD=1
      IFF=NFREQ
      GO TO 180
170  CONTINUE
      IFD=1
      IFF=25
C***** TRACE DU CADRE ET DU CARTOUCHE
180  CONTINUE
      XT=10.
      YT=200.
      ANG=0.
      NC=28
      CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
      KODE=1
      NCL=40
      NC=1
      CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
      KPLANC=KPLANC+1
C***** ECRITURE NASA ET TITRE LU SUR CARTE
      XT=CX-10.
      YT=CY-55.
      HL=3.
      HH=5.
      ANG=270.
      NC=19
      CALL KSTEP(1,1,4,1,XMOD)
      CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
C*****
      XT=10.
      YT=180.

```

```

HH=3.
ANG=0.
CALL KTEXTE(XT,YT,HL,HH,ITIT,ANG,NCAR,XMOD)
C***** ECRITURE DE NPOI, X, Y ET KMIC
XT=10.
YT=170.
NC=6
CALL KTEXTE(XT,YT,HL,HH,IPOINT,ANG,NC,XMOD)
XT=XT+18.
NC=4
CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
XT=XT+20.
NC=8
CALL KTEXTE(XT,YT,HL,HH,IX,ANG,NC,XMOD)
XT=XT+6.
NC=5
CALL KDECIM(XT,YT,HL,HH,X,ANG,NC,XMOD)
XT=XT+23.
NC=8
CALL KTEXTE(XT,YT,HL,HH,IY,ANG,NC,XMOD)
XT=XT+6.
NC=5
CALL KDECIM(XT,YT,HL,HH,Y,ANG,NC,XMOD)
XT=XT+23.
NC=8
CALL KTEXTE(XT,YT,HL,HH,NMIC,ANG,NC,XMOD)
XT=XT+27.
NC=1
CALL KFIXE(XT,YT,HL,HH,MICRO(1),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(2),ANG,NC,XMOD)
IF(KMIC.NE.0) GO TO 190
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(3),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(4),ANG,NC,XMOD)
190 CONTINUE
C***** ECRITURE 'FREQ.          A '
XT=20.
YT=155.
NC=5
CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
XT=XT+42.
NC=1
CALL KTEXTE(XT,YT,HL,HH,IA,ANG,NC,XMOD)
XT=XT+30.
NC=2
CALL KTEXTE(XT,YT,HL,HH,IHZ,ANG,NC,XMOD)
C*****
XT=160.
NC=5
CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
XT=XT+42.
NC=1

```

```

CALL KTEXTE(XT,YT,HL,HH,1A,ANG,NC,XMOD)
XT=XT+30.
NC=2
CALL KTEXTE(XT,YT,HL,HH,1HZ,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL
XT=10.
YT=10.
NC=0
PASMM=10.
ANG=90.
XMIN=1.
PAS=1.
NGRAD=15
NF=0
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL
XT=150.
YT=10.
NGRAD=15
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE GAUCHE DE LA PLANCHE)
XT=10.
YT=10.
PASMM=7.5
ANG=0.
NGRAD=14
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE DROITE DE LA PLANCHE)
XT=150.
YT=10.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
YT=5.
HL=2.
HH=3.
NC=3
DO 200 I=1,4
XT=XIANG(I)
CALL KFIXE(XT,YT,HL,HH,1ANG(I),ANG,NC,XMOD)
200 CONTINUE
XT=XT+10.
NC=3
CALL KTEXTE(XT,YT,HL,HH,1DEG,ANG,NC,XMOD)
YT=15.
NC=6
CALL KTEXTE(XT,YT,HL,HH,1THETAS,ANG,NC,XMOD)
NC=3
C*****
YT=5.
DO 210 I=1,4
XT=XIANG(I+4)
CALL KFIXE(XT,YT,HL,HH,1ANG(I),ANG,NC,XMOD)
210 CONTINUE
XT=XT+10.
NC=3
CALL KTEXTE(XT,YT,HL,HH,1DEG,ANG,NC,XMOD)
YT=15.
NC=6
CALL KTEXTE(XT,YT,HL,HH,1THETAS,ANG,NC,XMOD)
NC=3

```

```

C***** ECRITURE 2DB
    XT=8.
    YT=142.
    HH=2.
    ANG=90.
    NC=3
    CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
    XT=148.
    YT=142.
    CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
C***** ECRITURE DES VALEURS MAX
    XT=12.
    YT=150.
    HH=2.
    HL=2.
    ANG=0.
    NC=5
    CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
    XT=152.
    CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
C***** ECRITURE DES FREQUENCES
    XT=38.
    YT=155.
    NC=7
    HL=3.
    HH=3.
    ANG=0.
    SFREQ=SNGL(FREQ(IFD))
    CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
    IF((IFD+9).GE.IFF) GO TO 240
    SFREQ=SNGL(FREQ(IFD+9))
    XT=XT+30.
    CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C*****
    XT=178.
    SFREQ=SNGL(FREQ(IFD+10))
    CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
240 CONTINUE
    XT=XT+30.
    SFREQ=SNGL(FREQ(1FF))
    CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C***** TRACE DES COURBES
    CXSUJ1=40.
    CXSUJ2=170.
    CYSUJ1=DMIN
    CYSUJ2=DMAX
    CXOBJ1=10.
    CXOBJ2=107.5
    CYOBJ1=10.
    CYOBJ2=150.
    IDEB=IFD
    IF(1FF.GT.(IFD+9)) GO TO 280
    IFIN=1FF
    GO TO 290
280 CONTINUE
    IFIN=IFD+9
290 CONTINUE
C*****
    CALL DEPLAC(0.,XMOD)
    CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)

```

```

      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
      K=0
      DO 340 I=IDEB,IFIN
      DO 320 J=1,MIKEA
      DIFF(J)=SNGL(DECB(J,1))
320  CONTINUE
C*****
330  CONTINUE
      K=K+1
      IF(K.GT.15) WRITE(108,500) ; RETURN
      CALL KSTEP(1,NTR(K),NEP(K),NPL(K),XMOD)
      CALL KTRACE(SANG,DIFF,NBPT,XMOD)
340  CONTINUE
C*****
      CALL KSTEP(1,1,1,1,XMOD)
      IF(IFIN.EQ.IFF) GO TO 350
      IDEB=IFIN+1
      IFIN=IFF
      CXOBJ1=150.
      CXOBJ2=247.5
      GO TO 290
C*****
350  CONTINUE
      IF(IFF.EQ.NFREQ) GO TO 360
      IFD=IFF+1
      IFF=NFREQ
      CCX=CCX+20.
      CALL DEPLAC(CCX,XMOD)
      GO TO 180
360  CONTINUE
      CCX=CCX+20.
      CALL DEPLAC(CCX,XMOD)
C*****
      RETURN
500  FORMAT(1X,'TROP DE COURBES 1/3 OCTAVES '//)
      END
*DECK TDECB2
      SUBROUTINE TDECB2(KPLANC,DECB,ANGLES,PSIVAL,FREQ,NFREQ,MIKEA,YY,
      1ITIT,NCAR,XMOD)
C*****
C***** SOUS-PROGRAMME DE TRACE DES COURBES DECB=F(ANGLES)
C*****
      REAL*8 DECB(50,35),ANGLES(50,35),FREQ(33),YY,PSIVAL(50)
      COMMON /IDENT/NP01,KMIC,NP01F,KLIO,IOLDEB,IOLFIN
      COMMON /TITRE/LIB(10),IDATE(3),DB1(8,4),IP1(8),LIB1(20),LIB2(20)
      DIMENSION ITIT(1),XMOD(1)
C*****
      DIMENSION SANG(50)
      DIMENSION NAM(7),NASA(5),IP0INT(2),IX(1),IY(3),NMIC(2),IDEG(1)
      DIMENSION MICRO(4),MOINS(1),IPSI2(1)
      DIMENSION IANG(4),XIANG(8),ISDB(1),IFREQ(2),IA(1),IHZ(1),IGLOBAL
      *(3),IHHZ(5)
      DIMENSION X12(2),Y12(2)
      DIMENSION DIFF(50)
      DIMENSION NTR(15),NEP(15),NPL(15)
C*****
      DATA CX/300./,CY/210./
      DATA NAM/'P9-COURATIN' ,PROG.NOISE4 '/'
      DATA NASA/'N.A.S.A./O.N.E.R.A.' '/'
      DATA IP0INT/'POINT' ,IX/'X=' ,IY/'Y=' M ,NMIC/' MICRO

```

```

*= '/,IPSI2/'PSI2'/
DATA IANG/50,90,130,170/
DATA XIANG/13.,43.,73.,103.,153.,183.,213.,243./
DATA IDEG/'DEG '/,15DB/'5DB '/,IFREQ/'FREQ. '/,IGLOBAL/'GLOBAL(P
*SIS)'/
DATA MOINS/'- '/,1HZ/'HZ '/,IA/'A '/
DATA 1HHZ/'( ' HZ) '/
DATA NTR/1,2,3,4,5,1,2,3,4,5,1,2,3,4,5/
DATA NEP/1,2,1,1,1,1,2,1,1,1,1,2,1,1,1/
DATA NPL/1,1,1,1,1,2,2,2,2,3,3,3,3,3/
GLDEB=SNGL(FREQ(IGLDEB))
GLFIN=SNGL(FREQ(IGLFIN))
C***** TRAITEMENT LIGNE 1
C***** INITIALISATION DES VALEURS POUR LA LIGNE 1
Y=SNGL(YY)/3.2808
DO 150 I=1,MIKEA
SANG(I)=SNGL(PSIVAL(I))
DIFF(I)=SNGL(DECB(I,34))
150 CONTINUE
NBPT=MIKEA
C***** NOMBRE DE FREQUENCES PAR LIGNE
NU=1
CALL MINMAX(DIFF,NBPT,NU,DMIN,DMAX,XMOD)
IDMAX=(DMAX/10.)+1
IDMAX=IDMAX*10
IDMIN=IDMAX-70
DMAX=IDMAX
DMIN=IDMIN
DO 190 III=1,NFREQ
IF(DECB(1,III).LT.1.) GO TO 190
IFI=III
GO TO 195
190 CONTINUE
195 IF(NFREQ.GT.25) GO TO 170
IFD=1
IFF=NFREQ
GO TO 180
170 CONTINUE
IFD=1
IFF=25
C***** TRACE DU CADRE , DU CARTOUCHE ET DU LIBELLE
180 CONTINUE
XT=10.
YT=200.
ANG=0.
NC=28
CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
XT=95.
YT=202.
NC=80
HL=2.5
HH=2.5
CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
YT=195.
CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
KODE=1
NCL=40
NC=1
CALL TCART(KODE,CX,CY,LIB,NCL,1DATE,NC,XMOD)
KPLANC=KPLANC+1

```

```

C***** ECRITURE NASA ET TITRE LU SUR CARTE
      XT=CX-10.
      YT=CY-55.
      HL=3.
      HH=5.
      ANG=270.
      NC=19
      CALL KSTEP(1,1,4,1,XMOD)
      CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
C*****
      XT=10.
      YT=180.
      HL=3.
      ANG=0.
      CALL KTEXTE(XT,YT,HL,HH,ITIT,ANG,NCAR,XMOD)
      YT=170.
      XT=50.
      NC=9
      CALL KTEXTE(XT,YT,HL,HH,IY,ANG,NC,XMOD)
      XT=XT+6.
      NC=6
      CALL KDECIM(XT,YT,HL,HH,Y,ANG,NC,XMOD)
C***** ECRITURE DU NUMERO DE POINT ET DE B DE F
      XT=100.
      YT=180.
      HH=3.
      HL=3.
      ANG=0.
      NC=6
      CALL KTEXTE(XT,YT,HL,HH,'POINT ',ANG,NC,XMOD)
      XT=130.
      NC=4
      CALL KFIXE(XT,YT,HL,HH,NP0I,ANG,NC,XMOD)
      XT=155.
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,'B DE F ',ANG,NC,XMOD)
      XT=180.
      NC=4
      CALL KFIXE(XT,YT,HL,HH,NP0IF,ANG,NC,XMOD)
C***** ECRITURE 'FREQ.      A '
      XT=20.
      YT=155.
      NC=5
      CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
      XT=XT+42.
      NC=1
      CALL KTEXTE(XT,YT,HL,HH,IA,ANG,NC,XMOD)
      XT=XT+30.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,IHZ,ANG,NC,XMOD)
C*****
      XT=160.
      NC=5
      CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
      XT=XT+42.
      NC=1
      CALL KTEXTE(XT,YT,HL,HH,IA,ANG,NC,XMOD)
      XT=XT+30.
      NC=2

```

```

      CALL KTEXTE(XT,YT,HL,HH,1HZ,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL
      XT=10.
      YT=10.
      NC=0
      PASMM=10.
      ANG=90.
      XMIN=1.
      PAS=1.
      NGRAD=15
      NF=0
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL
      XT=150.
      YT=10.
      NGRAD=15
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE GAUCHE DE LA PLANCHE)
      XT=10.
      YT=10.
      PASMM=7.5
      ANG=0.
      NGRAD=14
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE DROITE DE LA PLANCHE)
      XT=150.
      YT=10.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
      YT=5.
      HL=2.
      HH=3.
      NC=3
      DO 200 I=1,4
      XT=XIANG(I)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
200  CONTINUE
      XT=XT+10.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,1DEG,ANG,NC,XMOD)
      YT=15.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1PS12,ANG,NC,XMOD)
      NC=3
C*****
      YT=5.
      DO 210 I=1,4
      XT=XIANG(I+4)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
210  CONTINUE
      XT=XT+10.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,1DEG,ANG,NC,XMOD)
      YT=15.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1PS12,ANG,NC,XMOD)
C***** ECRITURE 2DB
      XT=8.
      YT=142.
      HH=2.

```



```

ANG=90.
NC=3
CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
XT=148.
YT=142.
CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
C***** ECRITURE DES VALEURS MAX
XT=12.
YT=150.
HH=2.
HL=2.
ANG=0.
NC=5
CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
XT=152.
CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
C***** ECRITURE DES FREQUENCES
XT=38.
YT=155.
NC=7
HL=3.
HH=3.
ANG=0.
SFREQ=SNGL(FREQ(IFI))
CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
IF((IFD+9).GE.IFF) GO TO 240
SFREQ=SNGL(FREQ(IFD+9))
XT=XT+30.
CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C***** ECRITURE GLOBAL
XT=102.
YT=140.
NC=12
CALL KTEXTE(XT,YT,HL,HH,1GLOBAL,ANG,NC,XMOD)
NC=20
YT=134.
HL=2.
HH=2.
CALL KTEXTE(XT,YT,HL,HH,1HHZ,ANG,NC,XMOD)
XT=104.
NC=7
CALL KDECIM(XT,YT,HL,HH,GLDEB,ANG,NC,XMOD)
XT=120.
CALL KDECIM(XT,YT,HL,HH,GLFIN,ANG,NC,XMOD)
HL=3.
HH=3.
C*****
XT=178.
YT=155.
SFREQ=SNGL(FREQ(IFD+10))
CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
240 CONTINUE
XT=XT+30.
SFREQ=SNGL(FREQ(IFF))
CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C***** TRACE DES COURBES
CXSUJ1=40.
CXSUJ2=170.
CYSUJ1=DMIN
CYSUJ2=DMAX

```

```

        CXOBJ1=10.
        CXOBJ2=107.5
        CYOBJ1=10.
        CYOBJ2=150.
        IDEB=IFD
        IF(IFF.GT.(IFD+9)) GO TO 280
        IFIN=IFF
        GO TO 290
280 CONTINUE
        IFIN=IFD+9
290 CONTINUE
C***** TRACE DU GLOBAL (PSIVAL)
        CALL DEPLAC(0.,XM0D)
        CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XM0D)
        CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XM0D)
        CALL KSTEP(1,1,1,1,XM0D)
        CALL KTRACE(SANG,DIFF,NBPT,XM0D)
C***** TRACE DES 1/3 D'OCTAVE (PS12)
390 CALL DEPLAC(0.,XM0D)
        CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XM0D)
        CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XM0D)
        K=0
        DO 340 I=IDEB,IFIN
        DO 320 J=1,MIKEA
            SANG(J)=SNGL(ANGLES(J,I))
            DIFF(J)=SNGL(DECB(J,I))
320 CONTINUE
C*****
330 CONTINUE
        K=K+1
        IF(K.GT.15) WRITE(108,500) ; RETURN
        CALL KSTEP(1,NTR(K),NEP(K),NPL(K),XM0D)
        CALL KTRACE(SANG,DIFF,NBPT,XM0D)
340 CONTINUE
C*****
        CALL KSTEP(1,1,1,1,XM0D)
        IF(IFIN.EQ.IFF) GO TO 350
        IDEB=IFIN+1
        IFIN=IFF
        CXOBJ1=150.
        CXOBJ2=247.5
        GO TO 390
C*****
350 CONTINUE
        IF(IFF.EQ.NFREQ) GO TO 360
        IFD=IFF+1
        IFF=NFREQ
        CCX=CX+20.
        CALL DEPLAC(CCX,XM0D)
        GO TO 180
360 CONTINUE
        CCX=CX+20.
        CALL DEPLAC(CCX,XM0D)
C*****
        RETURN
500 FORMAT(1X,'TROP DE COURBES 1/3 OCTAVES '//)
        END

```

```

*
CPROGRAMMES DES MISES EN BIBLIOTHEQUE BIBLØM
!JOB,T TPL,P003,P9CØURA,40,(REST) MISE EN BIBLØM DE TPL
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBLØM DU SOUS-PROGRAMME TPL
!LIMIT (CØRE,50),(TIME,1),(SPDISC,50),(PAGES,30)
!FORTRAN SI,LS,GØ
      SUBROUTINE TPL(CX,CY,X,Y,ANG,NAM,NC,XMØD)
C***** SOUS-PROGRAMME DE TRACE DU CADRE ET DU NOM
C***** CX=DIMENSION EN MM,SUIVANT L AXE DES ABSCISSES DE LA PLANCHE
C***** CY=DIMENSION EN MM,SUIVANT L AXE DES ORDØNNEES DE LA PLANCHE
C***** X,Y=COORDØNNEES UTILISATEUR(EN MM) ØU ØØIT DEBUTER LE TRACE
C*****          DU NOM
C***** ANG=VARIABLE REPRESENTANT L ANGLE EN DEGRES ENTRE L AXE
C*****          ØX DU TRACEUR ET LE SENS D ECRITURE DU NOM
C***** NC=NOMBRE DE CARACTERES DU NOM A ECRIRE
      DIMENSION XMØD(44)
      DIMENSION X12(5),Y12(5),NAM(1)
C*****
      DATA ZERO/0./
C*****
      X12(1)=ZERO
      X12(2)=ZERO
      X12(3)=CX
      X12(4)=CX
      X12(5)=ZERO
C*****
      Y12(1)=ZERO
      Y12(2)=CY
      Y12(3)=CY
      Y12(4)=ZERO
      Y12(5)=ZERO
C***** TRACE DU CADRE
      CALL DIMSUJ(ZERO,CX,ZERO,CY,XMØD)
      CALL DIMØBJ(ZERO,CX,ZERO,CY,XMØD)
      CALL KTRACE(X12,Y12,5,XMØD)
C***** TRACE DU NOM
      CALL KTEXTE(X,Y,3.,3.,NAM,ANG,NC,XMØD)
      RETURN
      END
!EXEC INØMB,%KEY=TPL
!JOB,T TCART,P003,P9CØURA,40,(REST) MISE EN BIBLØM DE TCART
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBLØM DU SOUS-PROGRAMME TCART
!LIMIT (CØRE,50),(TIME,1),(SPDISC,50),(PAGES,30)
!FORTRAN SI,LS,GØ
      SUBROUTINE TCART(KØDE,CX,CY,LIB,NCL,IDATE,NCD,XMØD)
C***** SOUS-PROGRAMME DE TRACE DE L EN-TETE ØNERA
C***** KØDE =1 SI EN-TETE TRACE PERPENDICULAIREMENT A L AXE ØX
C*****          DU TRACEUR
C*****          =0 SI EN-TETE TRACE PARALLELEMENT A L AXE ØX
C*****          DU TRACEUR
C***** CX = DIMENSION EN MM SUIVANT L AXE DES ABSCISSES DE LA
C*****          PLANCHE
C***** CY = DIMENSION EN MM SUIVANT L AXE DES ORDØNNEES DE LA
C*****          PLANCHE
C***** LIB =TABLEAU HØLLERITH CONTENANT LE TITRE A ECRIRE
C***** NCL = NOMBRE DE CARACTERES DU TITRE A ECRIRE
C*****          SI NCL=0 PAS D ECRITURE
C***** IDATE= TABLEAU CONTENANT LES VALEURS NUMERIQUES DE LA DATE

```

```

C*****      IDATE(1)=NUMERO DU JOUR
C*****      IDATE(2)=NUMERO DU MOIS
C*****      IDATE(3)=NUMERO DE L ANNEE
C***** NCD   =0 SI PAS D ECRITURE DE DATE
C*****
      DIMENSION LIB(1),IDATE(1)
      DIMENSION XMOD(44)
      DIMENSION X(2),Y(2),ION(3),IPA(1),S(1)
      DATA ION/'O.N.E.R.A.  '/,IPA/'PAGE'/,S/'/'/'/'
C***** MODIFICATION DE L EPAISSEUR DU TRAIT
      CALL KSTEP(1,1,4,1,XMOD)
C*****
      IF(KODE.EQ.1) GO TO 130
C***** TRACE DE L EN-TETE PARALLELEMENT A L AXE OX DU TRACEUR
      X(1)=20.
      X(2)=X(1)
      Y(1)=0.
      Y(2)=CY
      CALL KTRACE(X,Y,2,XMOD)
      X(2)=CX
      Y(1)=(CY-20.)
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=53.
      X(2)=X(1)
      Y(1)=(CY-20.)
      Y(2)=CY
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=(CX-30.)
      X(2)=X(1)
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=53.
      X(2)=CX
      Y(1)=(CY-13.)
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
C***** ECRITURE ONERA ET PAGE
      XA=21.
      YA=(CY-15.)
      CALL KTEXTE(XA,YA,3.,10.,ION,0.,10,XMOD)
      XA=(CX-28.)
      YA=(CY-10.)
      CALL KTEXTE(XA,YA,3.,3.,IPA,0.,4,XMOD)
C***** ECRITURE EVENTUELLE DU TITRE
      IF(NCL.EQ.0) GO TO 120
      HX=3.
100 CONTINUE
      XLON=HX*NCL
      XA=CX-83.
      IF(XLON.LT.XA) GO TO 110
      HX=HX-0.5
      IF(HX.LE.0.) GO TO 120
      GO TO 100
110 CONTINUE
      XA=(XA-XLON)/2.
      XA=(53.+XA)
      YA=(CY-19.)
      CALL KTEXTE(XA,YA,HX,3.,LIB,0.,NCL,XMOD)
120 CONTINUE
C***** ECRITURE EVENTUELLE DE LA DATE

```

```

      IF(NCD.EQ.0) CALL KSTEP(1,1,1,1,XMOD) ; RETURN
      XA=(CX-29.)
      YA=(CY-18.)
      CALL KFIXE(XA,YA,3.,3.,IDATE(1),0.,+2,XMOD)
      XA=XA+7.
      CALL KTEXTE(XA,YA,3.,3.,S,0.,1,XMOD)
      XA=XA+4.
      CALL KFIXE(XA,YA,3.,3.,IDATE(2),0.,+2,XMOD)
      XA=XA+7.
      CALL KTEXTE(XA,YA,3.,3.,S,0.,1,XMOD)
      XA=XA+4.
      CALL KFIXE(XA,YA,3.,3.,IDATE(3),0.,+2,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
      RETURN
C***** TRACE DE L EN-TETE PERPENDICULAIREMENT A L AXE OX DU TRACEUR
130 CONTINUE
      X(1)=0.
      X(2)=CX
      Y(1)=(CY-20.)
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=(CX-20.)
      X(2)=X(1)
      Y(1)=0.
      CALL KTRACE(X,Y,2,XMOD)
      X(2)=CX
      Y(1)=(CY-53.)
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
      Y(1)=30.
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=(CX-13.)
      X(2)=X(1)
      Y(1)=0.
      Y(2)=(CY-53.)
      CALL KTRACE(X,Y,2,XMOD)
C***** ECRITURE ONERA ET PAGE
      XA=(CX-15.)
      YA=(CY-21.)
      CALL KTEXTE(XA,YA,3.,10.,10N,270.,10,XMOD)
      XA=(CX-10.)
      YA=28.
      CALL KTEXTE(XA,YA,3.,3.,1PA,270.,4,XMOD)
C***** ECRITURE EVENTUELLE DU TITRE
      IF(NCL.EQ.0) GO TO 220
      HY=3.
200 CONTINUE
      YLON=HY*NCL
      YA=CY-83.
      IF(YLON.LT.YA) GO TO 210
      HY=HY-0.5
      IF(HY.LE.0.) GO TO 220
      GO TO 200
210 CONTINUE
      XA=(CX-19.)
      YA=(YA-YLON)/2.
      YA=(CY-53.-YA)
      CALL KTEXTE(XA,YA,HY,3.,LIB,270.,NCL,XMOD)
220 CONTINUE

```

```

C***** ECRITURE EVENTUELLE DE LA DATE
      IF(NCD.EQ.0) CALL KSTEP(1,1,1,1,XMOD) ; RETURN
      XA=(CX-18.)
      YA=29.
      CALL KFIXE(XA,YA,3.,3.,1,DATE(1),270.,+2,XMOD)
      YA=YA-7.
      CALL KTEXTE(XA,YA,3.,3.,S,270.,1,XMOD)
      YA=YA-4.
      CALL KFIXE(XA,YA,3.,3.,1,DATE(2),270.,+2,XMOD)
      YA=YA-7.
      CALL KTEXTE(XA,YA,3.,3.,S,270.,1,XMOD)
      YA=YA-4.
      CALL KFIXE(XA,YA,3.,3.,1,DATE(3),270.,+2,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
      RETURN
      END
!EXEC INOMB,XKEY=TCART
!JOB,T TDEC,P003,P9C0URA,40,(REST) MISE EN BIBL0M DE TDEC
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBL0M DE TDEC:TRACE DES C0URBES AVANT LISSAGE
!LIMIT (C0RE,50),(TIME,1),(SPDISC,50),(PAGES,49)
!EXEC DL0M,XKEY=TDEC
!FORTRAN SI,LS,G0
      SUBROUTINE TDEC(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,MIKEB,
1 ITIT,NCAR,XMOD)
C*****
C***** SOUS-PROGRAMME DE TRACE DES C0URBES DECB=F(ANGLES)
C*****
      REAL*8 DECB(130,35),ANGLES(130),FREQ(33)
      COMMON /IDENT/NP01,KMIC
      COMMON /TITRE/LIB(10),1,DATE(3),DB1(8,4),1,PI(8)
      DIMENSION ITIT(1),XMOD(1)
C*****
      DIMENSION SANG(130)
      DIMENSION NAM(7),NASA(5),1,POINT(2),IX(1),IY(1),NMIC(2),1,DEG(1)
      DIMENSION MICRO(4),M0INS(1)
      DIMENSION IANG(4),XIANG(8),15DB(1),1,FREQ(2),1A(1),1,HZ(1)
      DIMENSION X12(2),Y12(2)
      DIMENSION DIFF(130)
      DIMENSION NTR(15),NEP(15),NPL(15)
C*****
      DATA CX/300./,CY/210./
      DATA NAM/'PA-J.BRASSEUR   PROG.NOISE3  '/
      DATA NASA/'N.A.S.A./O.N.E.R.A.  '/
      DATA 1POINT/'POINT    '/,IX/'X=    '/,IY/'Y=    '/,NMIC/' MICRO=  '/
      DATA IANG/50,90,130,170/
      DATA XIANG/13.,43.,73.,103.,153.,183.,213.,243./
      DATA 1DEG/'DEG  '/,15DB/'5DB  '/,1,FREQ/'FREQ.  '/
      DATA M0INS/'-    '/,1,HZ/'HZ  '/,1A/'A    '/
      DATA NTR/1,2,3,4,5,1,2,3,4,5,1,2,3,4,5/
      DATA NEP/1,2,1,1,1,1,2,1,1,1,1,2,1,1,1/
      DATA NPL/1,1,1,1,1,1,2,2,2,2,3,3,3,3,3/
C***** TRAITEMENT D UNE LIGNE
      ILIGN=1
C***** INITIALISATION DES VALEURS X , Y , SANG ET MICRO
120 CONTINUE
      IF(ILIGN.EQ.1) G0 T0 140
C***** INITIALISATION DES VALEURS P0UR LA LIGNE 2
      X=DB1(1,PI(3),1)
      Y=DB1(1,PI(3),2)

```

```

      DO 130 I=1,MIKEB
      SANG(I)=SNGL(ANGLES(I+MIKEA))
      DIFF(I)=SNGL(DECB(I+MIKEA,34))
130  CONTINUE
      NBPT=MIKEB
      IF(KMIC.EQ.1) MICRO(1)=IPI(3) ; MICRO(2)=IPI(7) ; GO TO 160
      IF(KMIC.EQ.2) MICRO(1)=IPI(4) ; MICRO(2)=IPI(8) ; GO TO 160
      MICRO(1)=IPI(3)
      MICRO(2)=IPI(7)
      MICRO(3)=IPI(4)
      MICRO(4)=IPI(8)
      GO TO 160
C***** INITIALISATION DES VALEURS POUR LA LIGNE 1
140  CONTINUE
      X=DB1(IPI(1),1)
      Y=DB1(IPI(1),2)
      DO 150 I=1,MIKEA
      SANG(I)=SNGL(ANGLES(I))
      DIFF(I)=SNGL(DECB(I,34))
150  CONTINUE
      NBPT=MIKEA
      IF(KMIC.EQ.1) MICRO(1)=IPI(1) ; MICRO(2)=IPI(5) ; GO TO 160
      IF(KMIC.EQ.2) MICRO(1)=IPI(2) ; MICRO(2)=IPI(6) ; GO TO 160
      MICRO(1)=IPI(1)
      MICRO(2)=IPI(5)
      MICRO(3)=IPI(2)
      MICRO(4)=IPI(6)
C***** NOMBRE DE FREQUENCES PAR LIGNE
160  CONTINUE
      CALL MINMAX(DIFF,NBPT,1,DMIN,DMAX,XMOD)
      IDMAX=(DMAX/10.)+1
      IDMAX=IDMAX*10
      IDMIN=IDMAX-70
      DMAX=IDMAX
      DMIN=IDMIN
      IF(NFREQ.GT.25) GO TO 170
      IFD=1
      IFF=NFREQ
      GO TO 180
170  CONTINUE
      IFD=1
      IFF=25
C***** TRACE DU CADRE ET DU CARTOUCHE
180  CONTINUE
      XT=10.
      YT=200.
      ANG=0.
      NC=28
      CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
      KODE=1
      NCL=40
      NC=1
      CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
      KPLANC=KPLANC+1
C***** ECRITURE NASA ET TITRE LU SUR CARTE
      XT=CX-10.
      YT=CY-55.
      HL=3.
      HH=5.
      ANG=270.

```

```

NC=19
CALL KSTEP(1,1,4,1,XMOD)
CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
CALL KSTEP(1,1,1,1,XMOD)
C*****
XT=10.
YT=180.
HH=3.
ANG=0.
CALL KTEXTE(XT,YT,HL,HH,ITIT,ANG,NCAR,XMOD)
C***** ECRITURE DE NPOI, X, Y ET KMIC
XT=10.
YT=170.
NC=6
CALL KTEXTE(XT,YT,HL,HH,IPOINT,ANG,NC,XMOD)
XT=XT+18.
NC=4
CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
XT=XT+20.
NC=2
CALL KTEXTE(XT,YT,HL,HH,IX,ANG,NC,XMOD)
XT=XT+6.
NC=5
CALL KDECIM(XT,YT,HL,HH,X,ANG,NC,XMOD)
XT=XT+23.
NC=2
CALL KTEXTE(XT,YT,HL,HH,IY,ANG,NC,XMOD)
XT=XT+6.
NC=5
CALL KDECIM(XT,YT,HL,HH,Y,ANG,NC,XMOD)
XT=XT+23.
NC=8
CALL KTEXTE(XT,YT,HL,HH,NMIC,ANG,NC,XMOD)
XT=XT+27.
NC=1
CALL KFIXE(XT,YT,HL,HH,MICRO(1),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,MOTINS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(2),ANG,NC,XMOD)
IF(KMIC.NE.0) GO TO 190
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,MOTINS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(3),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,MOTINS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(4),ANG,NC,XMOD)
190 CONTINUE
C***** ECRITURE 'FREQ.          A '
XT=20.
YT=155.
NC=5
CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
XT=XT+42.
NC=1
CALL KTEXTE(XT,YT,HL,HH,IA,ANG,NC,XMOD)
XT=XT+30.
NC=2

```



```

      CALL KTEXTE(XT,YT,HL,HH,IHZ,ANG,NC,XMOD)
C*****
      XT=160.
      NC=5
      CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
      XT=XT+42.
      NC=1
      CALL KTEXTE(XT,YT,HL,HH,IA,ANG,NC,XMOD)
      XT=XT+30.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,IHZ,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL
      XT=10.
      YT=10.
      NC=0
      PASMM=10.
      ANG=90.
      XMIN=1.
      PAS=1.
      NGRAD=15
      NF=0
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL
      XT=150.
      YT=10.
      NGRAD=15
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE GAUCHE DE LA PLANCHE)
      XT=10.
      YT=10.
      PASMM=7.5
      ANG=0.
      NGRAD=14
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE DROITE DE LA PLANCHE)
      XT=150.
      YT=10.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
      YT=5.
      HL=2.
      HH=3.
      NC=3
      DO 200 I=1,4
      XT=XIANG(I)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
200 CONTINUE
      XT=XT+10.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
C*****
      YT=5.
      DO 210 I=1,4
      XT=XIANG(I+4)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
210 CONTINUE
      XT=XT+10.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
C***** ECRITURE 2DB

```

```

      XT=8.
      YT=142.
      HH=2.
      ANG=90.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
      XT=148.
      YT=142.
      CALL KTEXTE(XT,YT,HL,HH,15DB,ANG,NC,XMOD)
C***** ECRITURE DES VALEURS MAX
      XT=12.
      YT=150.
      HH=2.
      HL=2.
      ANG=0.
      NC=5
      CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
      XT=152.
      CALL KDECIM(XT,YT,HL,HH,DMAX,ANG,NC,XMOD)
C***** ECRITURE DES FREQUENCES
      XT=38.
      YT=155.
      NC=7
      HL=3.
      HH=3.
      ANG=0.
      SFREQ=SNGL(FREQ(IFD))
      CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
      IF((IFD+9).GE.IFF) GO TO 240
      SFREQ=SNGL(FREQ(IFD+9))
      XT=XT+30.
      CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C*****
      XT=178.
      SFREQ=SNGL(FREQ(IFD+10))
      CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
240  CONTINUE
      XT=XT+30.
      SFREQ=SNGL(FREQ(1FF))
      CALL KDECIM(XT,YT,HL,HH,SFREQ,ANG,NC,XMOD)
C***** TRACE DES COURBES
      CXSUJ1=40.
      CXSUJ2=170.
      CYSUJ1=DMIN
      CYSUJ2=DMAX
      CXOBJ1=10.
      CXOBJ2=107.5
      CYOBJ1=10.
      CYOBJ2=150.
      IDEB=IFD
      IF(1FF.GT.(IFD+9)) GO TO 280
      IFIN=1FF
      GO TO 290
280  CONTINUE
      IFIN=IFD+9
290  CONTINUE
C*****
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)

```

```

      K=0
      DO 340 I=1DEB,IFIN
      IF(ILIGN.EQ.1) GO TO 310
C*****
      DO 300 J=1,MIKEB
      DIFF(J)=SNGL(DECBS(J+MIKEA,1))
      300 CONTINUE
      GO TO 330
C*****
      310 CONTINUE
      DO 320 J=1,MIKEA
      DIFF(J)=SNGL(DECBS(J,1))
      320 CONTINUE
C*****
      330 CONTINUE
      K=K+1
      IF(K.GT.15) WRITE(108,500) ; RETURN
      CALL KSTEP(1,NTR(K),NEP(K),NPL(K),XM0D)
      CALL KTRACE(SANG,DIFF,NBPT,XM0D)
      340 CONTINUE
C*****
      CALL KSTEP(1,1,1,1,XM0D)
      IF(IFIN.EQ.1FF) GO TO 350
      IDEB=IFIN+1
      IFIN=1FF
      CX0BJ1=150.
      CX0BJ2=247.5
      GO TO 290
C*****
      350 CONTINUE
      IF(1FF.EQ.NFREQ) GO TO 360
      IFD=1FF+1
      1FF=NFREQ
      CCX=CCX+20.
      CALL DEPLAC(CCX,XM0D)
      GO TO 180
      360 CONTINUE
      ILIGN=ILIGN+1
      CCX=CCX+20.
      CALL DEPLAC(CCX,XM0D)
      IF(ILIGN.LE.2) GO TO 120
C*****
      RETURN
      500 FORMAT(1X,'TR0P DE COURBES 1/3 OCTAVES '//)
      END
IEXEC IN0MB,%KEY=TDECB
!JOB,T TDIF,P003,P9C0URA,40,(REST) MISE EN BIBL0M DE TDIFF
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBL0M DE TDIFF:TRACE DES COURBES DE DIFFERENCE
ILIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
IEXEC DL0M,%KEY=TDIFF
IF0RTAN SI,LS,00
      SUBROUTINE TDIFF(KPLANC,DECB,ANGLES,FREQ,NFREQ,MIKEA,MIKEB,
      1ITIT,NCAR,XM0D)
C*****
C***** SOUS-PROGRAMME DE TRACE DES DIFFERENCES
C*****
      REAL*8 DECB(130,35),ANGLES(130),FREQ(33)
      REAL*8 DECBS
      COMMON /DONTD/DECBS(130,35)

```

```

COMMON /IDENT/NP01,KMIC
COMMON /TITRE/LIB(10),IDATE(3),DB1(8,4),IPI(8)
DIMENSION ITIT(1),XMOD(1)
C*****
DIMENSION SFREQ(33),SANG(130)
DIMENSION NAM(7),NASA(5),IPOINT(2),IX(1),IY(1),NMIC(2),IDEG(1)
DIMENSION MICRO(4),MOINS(1)
DIMENSION IANG(4),XIANG(8),I2DB(1),IFREQ(1)
DIMENSION X12(2),Y12(2)
DIMENSION DIFF(130)
C*****
DATA CX/300./,CY/210./
DATA NAM/'PA-J.BRASSEUR   PROG.NOISE3 '/
DATA NASA/'N.A.S.A./O.N.E.R.A. '/
DATA IPOINT/'POINT   ',IX/'X=   ',IY/'Y=   ',NMIC/' MICRO= '/
DATA IANG/50,90,130,170/
DATA XIANG/13.,43.,73.,103.,153.,183.,213.,243./
DATA IDEG/'DEG   ',I2DB/'2DB   ',IFREQ/'FREQ'/
DATA MOINS/'-   '/
C***** CALCUL DES DIFFERENCES EN DOUBLE PRECISION
DO 100 J=1,130
DO 100 I=1,35
DECBS(J,I)=DECBS(J,I)-DECB(J,I)
100 CONTINUE
C***** CONVERSION DES FREQUENCES EN SIMPLE PRECISION
DO 110 I=1,33
SFREQ(I)=SNGL(FREQ(I))
110 CONTINUE
C***** TRAITEMENT D UNE LIGNE
ILIGN=1
C***** INITIALISATION DES VALEURS X , Y , SANG ET MICRO
120 CONTINUE
IF(ILIGN.EQ.1) GO TO 140
C***** INITIALISATION DES VALEURS POUR LA LIGNE 2
X=DB1(IPI(3),1)
Y=DB1(IPI(3),2)
DO 130 I=1,MIKEB
SANG(I)=SNGL(ANGLES(I+MIKEA))
130 CONTINUE
NBPT=MIKEB
IF(KMIC.EQ.1) MICRO(1)=IPI(3) ; MICRO(2)=IPI(7) ; GO TO 160
IF(KMIC.EQ.2) MICRO(1)=IPI(4) ; MICRO(2)=IPI(8) ; GO TO 160
MICRO(1)=IPI(3)
MICRO(2)=IPI(7)
MICRO(3)=IPI(4)
MICRO(4)=IPI(8)
GO TO 160
C***** INITIALISATION DES VALEURS POUR LA LIGNE 1
140 CONTINUE
X=DB1(IPI(1),1)
Y=DB1(IPI(1),2)
DO 150 I=1,MIKEA
SANG(I)=SNGL(ANGLES(I))
150 CONTINUE
NBPT=MIKEA
IF(KMIC.EQ.1) MICRO(1)=IPI(1) ; MICRO(2)=IPI(5) ; GO TO 160
IF(KMIC.EQ.2) MICRO(1)=IPI(2) ; MICRO(2)=IPI(6) ; GO TO 160
MICRO(1)=IPI(1)
MICRO(2)=IPI(5)
MICRO(3)=IPI(2)

```

```

      MICRO(4)=IPI(6)
C***** NOMBRE DE FREQUENCES PAR LIGNE
160 CONTINUE
      IF(NFREQ.GT.25) GO TO 170
      IFD=1
      IFF=NFREQ
      GO TO 180
170 CONTINUE
      IFD=1
      IFF=25
C***** TRACE DU CADRE ET DU CARTOUCHE
180 CONTINUE
      XT=10.
      YT=200.
      ANG=0.
      NC=28
      CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
      KODE=1
      NCL=40
      NC=1
      CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
      KPLANC=KPLANC+1
C***** ECRITURE NASA ET TITRE LU SUR CARTE
      XT=CX-10.
      YT=CY-55.
      HL=3.
      HH=5.
      ANG=270.
      NC=19
      CALL KSTEP(1,1,4,1,XMOD)
      CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
C*****
      XT=10.
      YT=180.
      HL=3.
      ANG=0.
      CALL KTEXTE(XT,YT,HL,HH,ITIT,ANG,NCAR,XMOD)
C***** ECRITURE DE NPOI, X, Y ET KMIC
      XT=10.
      YT=170.
      NC=6
      CALL KTEXTE(XT,YT,HL,HH,IPOINT,ANG,NC,XMOD)
      XT=XT+18.
      NC=4
      CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
      XT=XT+20.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,IX,ANG,NC,XMOD)
      XT=XT+6.
      NC=5
      CALL KDECIM(XT,YT,HL,HH,X,ANG,NC,XMOD)
      XT=XT+23.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,IY,ANG,NC,XMOD)
      XT=XT+6.
      NC=5
      CALL KDECIM(XT,YT,HL,HH,Y,ANG,NC,XMOD)
      XT=XT+23.
      NC=8

```

```

CALL KTEXTE(XT,YT,HL,HH,NMIC,ANG,NC,XMOD)
XT=XT+27.
NC=1
CALL KFIXE(XT,YT,HL,HH,MICRO(1),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(2),ANG,NC,XMOD)
IF(KMIC.NE.0) GO TO 190
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(3),ANG,NC,XMOD)
XT=XT+6.
CALL KTEXTE(XT,YT,HL,HH,M0INS,ANG,NC,XMOD)
XT=XT+6.
CALL KFIXE(XT,YT,HL,HH,MICRO(4),ANG,NC,XMOD)
190 CONTINUE
C***** TRACE DU PREMIER AXE VERTICAL
XT=10.
YT=40.
NC=0
PASMM=10.
ANG=90.
XMIN=1.
PAS=1.
NGRAD=11
NF=0
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL
XT=150.
YT=10.
NGRAD=16
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE GAUCHE DE LA PLANCHE)
XT=10.
YT=45.
PASMM=7.5
ANG=0.
NGRAD=14
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE PREMIER AXE HORIZONTAL(PARTIE DROITE DE LA PLANCHE)
XT=150.
YT=15.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
YT=35.
HL=2.
HH=3.
NC=3
DO 200 I=1,4
XT=XIANG(I)
CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
200 CONTINUE
XT=XT+10.
NC=3
CALL KTEXTE(XT,YT,HL,HH,IDE0,ANG,NC,XMOD)
C*****
YT=5.
DO 210 I=1,4

```

```

        XT=XIANG(1+4)
        CALL KFIXE(XT,YT,HL,HH,IANG(1),ANG,NC,XMOD)
210  CONTINUE
        XT=XT+10.
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,1DEG,ANG,NC,XMOD)
C***** ECRITURE 2DB
        XT=8.
        YT=132.
        HH=2.
        ANG=90.
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,12DB,ANG,NC,XMOD)
        XT=148.
        YT=152.
        CALL KTEXTE(XT,YT,HL,HH,12DB,ANG,NC,XMOD)
C***** TRACE DES AUTRES AXES
        X12(1)=10.
        X12(2)=107.5
        Y12(1)=45.
        DO 220 I=1,9
        Y12(1)=Y12(1)+10.
        Y12(2)=Y12(1)
        CALL KTRACE(X12,Y12,2,XMOD)
220  CONTINUE
C*****
        X12(1)=150.
        X12(2)=247.5
        Y12(1)=15.
        DO 230 I=1,14
        Y12(1)=Y12(1)+10.
        Y12(2)=Y12(1)
        CALL KTRACE(X12,Y12,2,XMOD)
230  CONTINUE
C***** ECRITURE DES FREQUENCES EN BOUT D AXE
        YT=45.
        XT=111.
        HL=2.
        HH=2.
        ANG=0.
        NC=7
        DO 240 I=1FD,1FD+9
        IF(1.GT.IFF) GO TO 260
        CALL KDECIM(XT,YT,HL,HH,SFREQ(1),ANG,NC,XMOD)
        YT=YT+10.
240  CONTINUE
        YT=YT-5.
        NC=4
        CALL KTEXTE(XT,YT,HL,HH,1FREQ,ANG,NC,XMOD)
C*****
        XT=251.
        YT=15.
        NC=7
        DO 250 I=1FD+10,IFF
        CALL KDECIM(XT,YT,HL,HH,SFREQ(1),ANG,NC,XMOD)
        YT=YT+10.
250  CONTINUE
        YT=YT-5.
        NC=4
        CALL KTEXTE(XT,YT,HL,HH,1FREQ,ANG,NC,XMOD)

```

```

      GO TO 270
260 CONTINUE
      YT=YT-5.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
270 CONTINUE
C***** TRACE DES COURBES DE DIFFERENCES
      CXSUJ1=40.
      CXSUJ2=170.
      CYSUJ1=-1.
      CYSUJ2=+1.
      CXOBJ1=10.
      CXOBJ2=107.5
      CYOBJ1=30.
      CYOBJ2=40.
      IDEB=IFD
      IF(1FF.GT.(IFD+9)) GO TO 280
      IFIN=1FF
      GO TO 290
280 CONTINUE
      IFIN=IFD+9
290 CONTINUE
C*****
      DO 340 I=IDEB,IFIN
      IF(ILIGN.EQ.1) GO TO 310
C*****
      DO 300 J=1,MIKEB
      DIFF(J)=SNGL(DECBS(J+MIKEA,I))
300 CONTINUE
      GO TO 330
C*****
310 CONTINUE
      DO 320 J=1,MIKEA
      DIFF(J)=SNGL(DECBS(J,I))
320 CONTINUE
C*****
330 CONTINUE
      CYOBJ1=CYOBJ1+10.
      CYOBJ2=CYOBJ2+10.
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
      CALL KTRACE(SANG,DIFF,NBPT,XMOD)
340 CONTINUE
C*****
      IF(IFIN.EQ.1FF) GO TO 350
      IDEB=IFIN+1
      IFIN=1FF
      CXOBJ1=150.
      CXOBJ2=247.5
      CYOBJ1=0.
      CYOBJ2=10.
      GO TO 290
C*****
350 CONTINUE
      IF(1FF.EQ.NFREQ) GO TO 360
      IFD=1FF+1
      1FF=NFREQ
      CCX=CCX+20.
      CALL DEPLAC(CCX,XMOD)

```



```

      GO TO 180
360 CONTINUE
      ILIGN=ILIGN+1
      CCX=CCX+20.
      CALL DEPLAC(CCX,XMOD)
      IF(ILIGN.LE.2) GO TO 120
C*****
      RETURN
      END
      IEXEC INOMB,XKEY=TDIFF
      !JOB,T TXSD,POO3,P9COURA,40,(REST) MISE EN BIBLON DE TXSD
      !COMMENT ETUDE=2895PN141P
      !COMMENT MISE EN BIBLON DE TXSD:TRACE DES COURBES X/D=F(THETA-S)
      ILIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
      IEXEC DLOM,XKEY=TXSD
      IFORTRAN SI,LS,GO
          SUBROUTINE TXSD(KPLANC,XOVERD,THETAS,NTAB,NFREQ,XDMAX,
      1THETSM,RNZDIA,VJET,FREQ,LIB1,LIB2,XMOD)
C***** SOUS-PROGRAMME DE TRACE DES COURBES X/D=F(THETAS)
      REAL*8 XOVERD(50,33),THETAS(50,33),XDMAX(33),THETSM(33)
      REAL*8 RNZDIA,VJET,FREQ(33)
      COMMON /IDENT/NP01
      COMMON /TITRE/LIB(10),IDATE(3)
      COMMON /TABD/STR(33),DFMAX(33),XPEAKN(33),XPEAKF(33)
      REAL*8 STR,DFMAX,XPEAKN,XPEAKF
      DIMENSION XMOD(1),NTAB(1),LIB1(1),LIB2(1)
C*****
      DIMENSION NAM(7),NASA(5),IP0INT(2),IDEG(1)
      DIMENSION IV(4),IDIA(3),IST(1),IDIF(2),IPEAK(3),INEAR(1),IFAR(1)
      DIMENSION ITHE(2),IXSD(1),IFREQ(1)
      DIMENSION IANG(7),XIANG(7),IVALX(6)
      DIMENSION SXSD(50),STHETA(50)
      DIMENSION IFIG(1)
      DIMENSION X12(2),Y12(2)
C*****
      DATA NAM/'PA-J.BRASSEUR   PROG.NOISE3 '/
      DATA NASA/'N.A.S.A./O.N.E.R.A. '/
      DATA IP0INT/'POINT   '/
      DATA IDEG/'DEG   '/
      DATA IV/'VJET(FT/SEC)=   '/
      DATA IDIA/'DIA.(FT)=   '/,IST/'ST=   '/,IDIF/'DIF. MAX'/
      DATA IPEAK/'PEAK(DEG.)   '/,INEAR/'NEAR'/,IFAR/'FAR   '/
      DATA ITHE/'THETA-S   '/,IXSD/'X/D   '/,IFREQ/'FREQ'/
C*****
      DATA CX/210./,CY/300./
      DATA IANG/40,60,80,100,120,140,160/
      DATA XIANG/47.,67.,87.,107.,127.,147.,167./
      DATA IVALX/0,2,4,6,8,10/
      DATA IFIG/'FIG.'/
C*****
C***** TRACE DE 2 COURBES PAR PLANCHE
      NBCT=0
      DO 210 IF=1,NFREQ
      IF(NTAB(IF).EQ.0) GO TO 210
      IF(NBCT.EQ.1) GO TO 160
C***** TRACE DU CADRE ET DU CARTOUCHE
      IF(NBCT.EQ.2) NBCT=0
      XT=10.
      YT=10.
      ANG=90.

```

```

NC=28
CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
KODE=0
NCL=40
NC=1
CALL TCART(KODE,CX,CY,LIB,NCL,DATE,NC,XMOD)
KPLANC=KPLANC+1
C***** ECRITURE NASA
XT=55.
YT=CY-10.
HL=3.
HH=5.
ANG=0.
NC=19
CALL KSTEP(1,1,4,1,XMOD)
CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
CALL KSTEP(1,1,1,1,XMOD)
C***** ECRITURE DANS LA MARGE DE LIB1 ET LIB2
HL=2.5
HH=2.5
NC=80
ANG=90.
XT=6.5
YT=100.
CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
XT=16.5
CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
C***** ECRITURE NPOI
XT=25.
YT=CY-28.
NC=5
HL=3.
ANG=0.
HH=3.
CALL KTEXTE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
C*****
XT=XT+18.
NC=4
CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
C***** ECRITURE VJET
XT=XT+18.
NC=13
CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
C*****
S=SNGL(VJET)
XT=XT+42.
NC=6
CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE DIA. (FT)
XT=XT+24.
NC=9
CALL KTEXTE(XT,YT,HL,HH,IDIA,ANG,NC,XMOD)
C*****
XT=XT+30.
NC=5
S=SNGL(RNZDIA)
CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE FIG.
XT=XT+30.
NC=4

```

```

      CALL KTEXTE(XT,YT,HL,HH,IFIG,ANG,NC,XMOD)
C***** TRACE DU TABLEAU
      X12(1)=20.
      X12(2)=CX-10.
      Y12(1)=CY-45.
      Y12(2)=Y12(1)
      CALL KTRACE(X12,Y12,2,XMOD)
      X12(1)=18.
      Y12(1)=CY-60.
      Y12(2)=CY-35.
      DO 110 I=1,7
      X12(1)=X12(1)+26.
      X12(2)=X12(1)
      IF(I.EQ.4) GO TO 100
      CALL KTRACE(X12,Y12,2,XMOD)
      GO TO 110
100 CONTINUE
      Y12(2)=Y12(2)-5.
      CALL KTRACE(X12,Y12,2,XMOD)
      Y12(2)=CY-35.
110 CONTINUE
C***** DIFFERENTES INSCRIPTIONS DU TABLEAU
      YT=CY-40.
      XT=26.
      ANG=0.
      HL=3.
      HH=3.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,IFREQ,ANG,NC,XMOD)
C*****
      XT=54.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C*****
      XT=72.
      NC=8
      CALL KTEXTE(XT,YT,HL,HH,IDIF,ANG,NC,XMOD)
C*****
      YT=YT+2.
      XT=107.
      NC=10
      CALL KTEXTE(XT,YT,HL,HH,IPEAK,ANG,NC,XMOD)
      YT=YT-5.
      XT=103.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,INEAR,ANG,NC,XMOD)
      XT=130.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IFAR,ANG,NC,XMOD)
C*****
      YT=CY-40.
      XT=150.
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
C*****
      XT=CX-28.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL(PARTIE HAUTE DE LA PLANCHE)
      XT=40.

```

```

YT=130.
NC=0
PASMM=20.
ANG=90.
XMIN=1.
PAS=1.
NGRAD=6
NF=0
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL(PARTIE BASSE DE LA PLANCHE)
YT=10.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU PREMIER AXE HORIZONTAL
XT=40.
YT=130.
ANG=0.
PASMM=10.
NGRAD=15
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE HORIZONTAL
YT=10.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
YT=5.
ANG=0.
HL=2.
HH=2.
NC=3
DO 120 I=1,7
XT=XIANG(I)
CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
120 CONTINUE
XT=XT+10.
NC=3
CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
YT=10.
XT=183.
NC=7
CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
YT=100.
XT=60.
NC=3
CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C*****
YT=125.
NC=3
DO 130 I=1,7
XT=XIANG(I)
CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
130 CONTINUE
XT=XT+10.
NC=3
CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
YT=YT+5.
XT=183.
NC=7
CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
YT=220.
XT=60.
NC=3

```

```

      CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES VERTICAUX
      XT=32.
      NC=2
      YT=-10.
      DO 140 I=1,6
      YT=YT+20.
      CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
140  CONTINUE
      XT=21.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C*****
      XT=32.
      NC=2
      DO 150 I=1,6
      YT=YT+20.
      CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
150  CONTINUE
      XT=21.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C***** CONVERSION EN SIMPLE PRECISION DE X/D ET THETA
160  CONTINUE
      DO 170 I=1,NTAB(IF)
      SXSD(I)=SNGL(XOVERD(I,IF))
      STHETA(I)=SNGL(THETAS(I,IF))
170  CONTINUE
      NBCT=NBCT+1
      IF(NBCT.EQ.1) YT=250. ; GO TO 180
C***** REMPLISSAGE DU TABLEAU
      YT=243.
180  CONTINUE
      XT=25.
      HL=2.
      HH=2.
      NC=7
      ANG=0.
      S=SNGL(FREQ(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=50.
      S=SNGL(STR(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(DFMAX(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(XPEAKN(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(XPEAKF(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(THETSM(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(XDMAX(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****
      IF(NBCT.EQ.1) YT=220. ; GO TO 190

```

```

      YT=100.
190 CONTINUE
      XT=70.
      NC=7
      S=SNGL(STR(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****
C***** TRACE EN POINTE DES COURBES
      CXSUJ1=30.
      CXSUJ2=170.
      CYSUJ1=-0.5
      CYSUJ2=10.
      CXOBJ1=40.
      CXOBJ2=CX-30.
      IF(NBCT.EQ.1) CYOBJ1=125. ; CYOBJ2=230. ; GO TO 200
      CYOBJ1=5.
      CYOBJ2=110.
200 CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
      NBPT=NTAB(IF)
      CALL KSTEP(-16,1,1,1,XMOD)
      CALL KPPOINT(STHETA, SXSD, NBPT, XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
      IF(NBCT.NE.2) GO TO 220
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      GO TO 210
220 CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(0.,CX,0.,CY,XMOD)
      CALL DIMOBJ(0.,CX,0.,CY,XMOD)
210 CONTINUE
      IF(NBCT.EQ.2) RETURN
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      RETURN
      END
IEXEC INOMB,%KEY=TXSD
!JOB,T TPEA,P003,P9COURA,40,(REST) MISE EN BIBLON DE TPEAK
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBLON DE TPEAK:TRACE DES COURBES THETA-S DU PIC=F(STROUHAL)
ILIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
IEXEC DLOM,%KEY=TPEAK
IFORTRAN SI,LS,GO
      SUBROUTINE TPEAK(KPLANC,THETSM,STR,NFREQ,RNZDIA,VJET,LIB1,LIB2
1,XMOD)
C***** SOUS-PROGRAMME DE TRACE DES COURBES THETSM=F(STR)
      REAL*8 THETSM(1),STR(1)
      REAL*8 RNZDIA,VJET
      DIMENSION XMOD(1),LIB1(1),LIB2(1)
C*****
      COMMON /IDENT/NP01
      COMMON /TITRE/LIB(10),IDATE(3)
C*****
      DIMENSION NAM(7),IPPOINT(2),IDIA(3),IV(4),IFIG(1)
      DIMENSION NASA(5)
      DIMENSION IVALY(6)
      DIMENSION ITHE(4),IST(2)

```

```

      DIMENSION X(2),Y(2)
      DIMENSION VALX(5),XX(5)
      DIMENSION STHETA(33),SSTR(33)
C*****
      DATA NAM/'PA-J.BRASSEUR   PROG.NOISE3 '/
      DATA CX/300./,CY/210./
      DATA IP0INT/'POINT   '/,IV/'VJET(FT/SEC)=   '/
      DATA IDIA/'DIA.(FT)=   '/,IFIG/'FIG.'/
      DATA NASA/'N.A.S.A./O.N.E.R.A. '/
      DATA IVALY/60,80,100,120,140,160/
      DATA ITHE/'THETA-S DU PIC '/
      DATA IST/'STROUHAL'/
      DATA VALX/+0.01,+0.10,+1.00,+10.0,+20.0/
      DATA XX/16.,86.,156.,226.,247./
C***** TRACE DU CADRE ET DU CARTOUCHE
      XT=10.
      YT=200.
      ANG=0.
      NC=28
      CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
      KODE=1
      NCL=40
      NC=1
      CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
      KPLANC=KPLANC+1
C***** ECRITURE NASA
      XT=CX-10.
      YT=CY-55.
      HL=3.
      HH=5.
      ANG=270.
      NC=19
      CALL KSTEP(1,1,4,1,XMOD)
      CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
C***** ECRITURE DANS LA MARGE DE LIB1 ET LIB2
      HL=2.5
      HH=2.5
      NC=80
      ANG=0.
      XT=100.
      YT=CY-6.5
      CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
      YT=CY-16.5
      CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
C***** ECRITURE DE POINT
      XT=10.
      YT=CY-30.
      NC=5
      HL=3.
      HH=3.
      ANG=0.
      CALL KTEXTE(XT,YT,HL,HH,IP0INT,ANG,NC,XMOD)
C*****
      XT=XT+18.
      NC=4
      CALL KFIXE(XT,YT,HL,HH,NP01,ANG,NC,XMOD)
C***** ECRITURE DE VJET
      XT=50.
      NC=13

```

```

      CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
C*****
      XT=XT+42.
      NC=6
      S=SNGL(VJET)
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE DIA.(FT)=
      XT=120.
      NC=9
      CALL KTEXTE(XT,YT,HL,HH,1DIA,ANG,NC,XMOD)
C*****
      S=SNGL(RNZDIA)
      XT=XT+30.
      NC=5
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE FIG.
      XT=200.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1FIG,ANG,NC,XMOD)
C***** TRACE DE L AXE VERTICAL
      XT=20.
      YT=20.
      ANG=90.
      XMIN=1.
      NC=0
      PAS=1.
      NGRAD=12
      NF=0
      PASMM=10.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR L AXE VERTICAL
      XT=10.
      HL=2.
      HH=2.
      YT=0.
      ANG=0.
      NC=3
      DO 100 I=1,6
      YT=YT+20.
      CALL KFIXE(XT,YT,HL,HH,IVALY(I),ANG,NC,XMOD)
100 CONTINUE
C***** ECRITURE DE THETA-S DU PIC
      YT=YT+13.
      XT=10.
      NC=14
      CALL KTEXTE(XT,YT,HL,HH,1THE,ANG,NC,XMOD)
C***** TRACE DE L AXE HORIZONTAL
      X(1)=20.
      X(2)=251.
      Y(1)=20.
      Y(2)=Y(1)
      CALL KTRACE(X,Y,2,XMOD)
C***** ECRITURE DE STROUHAL
      XT=255.
      YT=20.
      HL=2.
      HH=2.
      ANG=0.
      NC=8
      CALL KTEXTE(XT,YT,HL,HH,1ST,ANG,NC,XMOD)

```



```

C***** ECRITURE DES COTATIONS SUR L AXE HORIZONTAL
  YT=14.
  NC=5
  DO 110 I=1,5
    CALL KDECIM(XX(I),YT,HL,HH,VALX(I),ANG,NC,XMOD)
  110 CONTINUE
C***** GRADUATIONS DE L AXE HORIZONTAL
  CXOBJ1=20.
  CXOBJ2=251.
  CYOBJ1=17.5
  CYOBJ2=22.5
  CXSUJ1=ALOG10(0.01)
  CXSUJ2=ALOG10(20.)
  CYSUJ1=-2.5
  CYSUJ2=+2.5
  CALL DEPLAC(0.,XMOD)
  CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
  CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
C*****
  Y(1)=-2.5
  Y(2)=+2.5
  PASX=0.001
  DO 120 J=1,3
    PASX=PASX*10.
    DO 120 I=1,9
      X(1)=ALOG10(I*PASX)
      X(2)=X(1)
      CALL KTRACE(X,Y,2,XMOD)
    120 CONTINUE
      X(1)=ALOG10(10.)
      X(2)=X(1)
      CALL KTRACE(X,Y,2,XMOD)
      X(1)=ALOG10(20.)
      X(2)=X(1)
      CALL KTRACE(X,Y,2,XMOD)
C***** TRACE DES COURBES EN POINTE
  CXOBJ1=20.
  CXOBJ2=251.
  CYOBJ1=20.
  CYOBJ2=130.
  CXSUJ1=ALOG10(+0.01)
  CXSUJ2=ALOG10(+20.)
  CYSUJ1=60.
  CYSUJ2=170.
  CALL DEPLAC(0.,XMOD)
  CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
  CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
C***** CONVERSION EN SIMPLE PRECISION DE THETSM ET STR
  NBPT=0
  DO 130 I=1,NFREQ
    IF(STR(I).LT.(0.01).OR.STR(I).GT.(+20.)) GO TO 130
    NBPT=NBPT+1
    STHETA(NBPT)=SNGL(THETSM(I))
    SSTR(NBPT)=SNGL(STR(I))
    SSTR(NBPT)=ALOG10(SSTR(NBPT))
  130 CONTINUE
C*****
  CALL KSTEP(-16,1,1,1,XMOD)
  CALL KPOINT(SSTR,STHETA,NBPT,XMOD)
  CALL KSTEP(1,1,1,1,XMOD)

```

```

C*****
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      RETURN
      END
      IEXEC INOMB,%KEY=TPEAK
      !JOB,T DECB,P003,P9COURA,40,(REST) MISE EN BIBLON DE SAVDECB
      !COMMENT ETUDE=2895PN141P
      !COMMENT MISE EN BIBLON DU SOUS-PROGRAMME SAVDECB
      !LIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
      IEXEC DLON,%KEY=SAVDECB
      IFORTRAN SI,LS,LO,GO
          SUBROUTINE SAVDECB(DECB)
              REAL*8 DECB(130,35),DECBS
              COMMON /DONTD/DECBS(130,35)
              DO 100 J=1,130
                  DO 100 I=1,35
                      DECBS(J,I)=DECB(J,I)
              100 CONTINUE
              RETURN
          END
      IEXEC INOMB,%KEY=SAVDECB
      !JOB,T XSDN,P003,P9COURA,40,(REST) MISE EN BIBLON DE TXSDN
      !COMMENT ETUDE=2895PN141P
      !COMMENT MISE EN BIBLON DE TXSDN: TRACE DES COURBES X/D=F(THETA-S)
      !COMMENT AUX ECHELLES NASA
      !LIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
      IEXEC DLON,%KEY=TXSDN
      IFORTRAN SI,LS,GO
          SUBROUTINE TXSDN(KPLANC,XOVERD,THETAS,NTAB,NFREQ,XDMAX,
              1THETSM,RNZDIA,VJET,FREQ,LIB1,LIB2,XMOD)
C***** SOUS-PROGRAMME DE TRACE DES COURBES X/D=F(THETAS)
              REAL*8 XOVERD(50,33),THETAS(50,33),XDMAX(33),THETSM(33)
              REAL*8 RNZDIA,VJET,FREQ(33)
              COMMON /IDENT/NP01
              COMMON /TITRE/LIB(10),IDATE(3)
              COMMON /TABD/STR(33),DFMAX(33),XPEAKN(33),XPEAKF(33)
              REAL*8 STR,DFMAX,XPEAKN,XPEAKF
              DIMENSION XMOD(1),NTAB(1),LIB1(1),LIB2(1)
C*****
              DIMENSION NAM(7),NASA(5),IP0INT(2),IDEG(1)
              DIMENSION IV(4),IDIA(3),IST(1),IDIF(2),IPEAK(3),INEAR(1),IFAR(1)
              DIMENSION ITHE(2),IXSD(1),IFREQ(1)
              DIMENSION IANG(7),XIANG(7),IVALX(6)
              DIMENSION SXSD(50),STHETA(50)
              DIMENSION IFIG(1)
              DIMENSION X12(2),Y12(2)
C*****
              DATA NAM/'PA-J.BRASSEUR ECH.N.A.S.A.'/
              DATA NASA/'N.A.S.A./O.N.E.R.A.'/
              DATA IP0INT/'POINT '/
              DATA IDEG/'DEG '/
              DATA IV/'VJET(FT/SEC)= '/
              DATA IDIA/'DIA.(FT)= '/,IST/'ST= '/,IDIF/'DIF. MAX'/
              DATA IPEAK/'PEAK(DEG.) '/,INEAR/'NEAR'/,IFAR/'FAR '/
              DATA ITHE/'THETA-S '/,IXSD/'X/D '/,IFREQ/'FREQ'/
C*****
              DATA CX/210./,CY/300./
              DATA IANG/40,60,80,100,120,140,160/
              DATA XIANG/47.25,67.75,88.25,108.75,129.25,149.75,170.25/

```

```

DATA IVALX/0,2,4,6,8,10/
DATA IFIG/'FIG.'/
C*****
C***** TRACE DE 2 COURBES PAR PLANCHE
NBCT=0
DO 210 IF=1,NFREQ
IF(NTAB(IF).EQ.0) GO TO 210
IF(NBCT.EQ.1) GO TO 160
C***** TRACE DU CADRE ET DU CARTOUCHE
IF(NBCT.EQ.2) NBCT=0
XT=10.
YT=10.
ANG=90.
NC=28
CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
KODE=0
NCL=40
NC=1
CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
KPLANC=KPLANC+1
C***** ECRITURE NASA
XT=55.
YT=CY-10.
HL=3.
HH=5.
ANG=0.
NC=19
CALL KSTEP(1,1,4,1,XMOD)
CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
CALL KSTEP(1,1,1,1,XMOD)
C***** ECRITURE DANS LA MARGE DE LIB1,LIB2
HL=2.5
HH=2.5
NC=80
ANG=90.
XT=6.5
YT=100.
CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
XT=16.5
CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
C***** ECRITURE NPOI
XT=25.
YT=CY-28.
NC=5
HL=3.
HH=3.
ANG=0.
CALL KTEXTE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
C*****
XT=XT+18.
NC=4
CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
C***** ECRITURE VJET
XT=XT+18.
NC=13
CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
C*****
S=SNGL(VJET)
XT=XT+42.
NC=6

```

```

      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE DIA.(FT)
      XT=XT+24.
      NC=9
      CALL KTEXTE(XT,YT,HL,HH,1DIA,ANG,NC,XMOD)
C*****
      XT=XT+30.
      NC=5
      S=SNGL(RNZDIA)
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE FIG.
      XT=XT+30.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1FIG,ANG,NC,XMOD)
C***** TRACE DU TABLEAU
      X12(1)=20.
      X12(2)=CX-10.
      Y12(1)=CY-45.
      Y12(2)=Y12(1)
      CALL KTRACE(X12,Y12,2,XMOD)
      X12(1)=18.
      Y12(1)=CY-60.
      Y12(2)=CY-35.
      DO 110 I=1,7
      X12(1)=X12(1)+26.
      X12(2)=X12(1)
      IF(I.EQ.4) GO TO 100
      CALL KTRACE(X12,Y12,2,XMOD)
      GO TO 110
100 CONTINUE
      Y12(2)=Y12(2)-5.
      CALL KTRACE(X12,Y12,2,XMOD)
      Y12(2)=CY-35.
110 CONTINUE
C***** DIFFERENTES INSCRIPTIONS DU TABLEAU
      YT=CY-40.
      XT=26.
      ANG=0.
      HL=3.
      HH=3.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1FREQ,ANG,NC,XMOD)
C*****
      XT=54.
      NC=2
      CALL KTEXTE(XT,YT,HL,HH,1ST,ANG,NC,XMOD)
C*****
      XT=72.
      NC=8
      CALL KTEXTE(XT,YT,HL,HH,1DIF,ANG,NC,XMOD)
C*****
      YT=YT+2.
      XT=107.
      NC=10
      CALL KTEXTE(XT,YT,HL,HH,1PEAK,ANG,NC,XMOD)
      YT=YT-5.
      XT=103.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,1NEAR,ANG,NC,XMOD)
      XT=130.

```

```

      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IFAR,ANG,NC,XMOD)
C*****
      YT=CY-40.
      XT=150.
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
C*****
      XT=CX-28.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL(PARTIE HAUTE DE LA PLANCHE)
      XT=40.
      YT=130.
      NC=0
      PASMM=20.5
      ANG=90.
      XMIN=1.
      PAS=1.
      NGRAD=6
      NF=0
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL(PARTIE BASSE DE LA PLANCHE)
      YT=10.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU PREMIER AXE HORIZONTAL
      XT=40.
      YT=130.
      ANG=0.
      PASMM=10.25
      NGRAD=15
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE HORIZONTAL
      YT=10.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
      YT=5.
      ANG=0.
      HL=2.
      HH=2.
      NC=3
      DO 120 I=1,7
      XT=XIANG(I)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
120 CONTINUE
      XT=XT+10.25
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
      YT=10.
      XT=186.5
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
      YT=100.
      XT=60.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C*****
      YT=125.
      NC=3
      DO 130 I=1,7

```

```

        XT=XIANG(I)
        CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
130  CONTINUE
        XT=XT+10.25
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,IDEQ,ANG,NC,XMOD)
        YT=YT+5.
        XT=186.5
        NC=7
        CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
        YT=220.
        XT=60.
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES VERTICAUX
        XT=32.
        NC=2
        YT=-10.5
        DO 140 I=1,6
        YT=YT+20.5
        CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
140  CONTINUE
        XT=21.
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C*****
        XT=32.
        YT=109.5
        NC=2
        DO 150 I=1,6
        YT=YT+20.5
        CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
150  CONTINUE
        XT=21.
        NC=3
        CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C***** CONVERSION EN SIMPLE PRECISION DE X/D ET THETA
160  CONTINUE
        DO 170 I=1,NTAB(IF)
        SXSD(I)=SNGL(XOVERD(I,IF))
        STHETA(I)=SNGL(THETAS(I,IF))
170  CONTINUE
        NBCT=NBCT+1
        IF(NBCT.EQ.1) YT=250. ; GO TO 180
C***** REMPLISSAGE DU TABLEAU
        YT=243.
180  CONTINUE
        XT=25.
        HL=2.
        HH=2.
        NC=7
        ANG=0.
        S=SNGL(FREQ(IF))
        CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
        XT=50.
        S=SNGL(STR(IF))
        CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
        XT=XT+26.
        S=SNGL(DFMAX(IF))
        CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)

```

```

      XT=XT+26.
      S=SNGL(XPEAKN(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(XPEAKF(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(THETSM(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      XT=XT+26.
      S=SNGL(XDMAX(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****
      IF(NBCT.EQ.1) YT=220. ; GO TO 190
      YT=100.
190  CONTINUE
      XT=70.
      NC=7
      S=SNGL(STR(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****
C***** TRACE EN POINTE DES COURBES
      CXSUJ1=30.
      CXSUJ2=170.
      CYSUJ1=-0.5
      CYSUJ2=10.
      CXOBJ1=40.
      CXOBJ2=183.5
      IF(NBCT.EQ.1) CYOBJ1=124.875 ; CYOBJ2=232.5 ; GO TO 200
      CYOBJ1=4.875
      CYOBJ2=112.5
200  CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
      NBPT=NTAB(IF)
      CALL KSTEP(-16,1,1,1,XMOD)
      CALL KPPOINT(STHETA,SXSD,NBPT,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
      IF(NBCT.NE.2) GO TO 220
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      GO TO 210
220  CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(0.,CX,0.,CY,XMOD)
      CALL DIMOBJ(0.,CX,0.,CY,XMOD)
210  CONTINUE
      IF(NBCT.EQ.2) RETURN
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      RETURN
      END
IEXEC INOMB,%KEY=TXSDN
!JOB,T PEAN,P003,P9COURA,40,(REST) MISE EN BIBLON DE TPEAKN
!COMMENT ETUDE=2895PN141P
!COMMENT MISE EN BIBLON DE TPEAKN:TRACE DES COURBES THETA-S DU PIC=F(STROUHAL)
!COMMENT AUX ECHELLES NASA
!LIMIT (CORE,50),(TIME,1),(SPDISC,50),(PAGES,49)
IEXEC DLON,%KEY=TPEAKN

```

```

IFORTRAN SI,LS,GO
SUBROUTINE TPEAKN(KPLANC,THETSM,STR,NFREQ,RNZDIA,VJET,LIB1,LIB2
1,XMOD)
C***** SOUS-PROGRAMME DE TRACE DES COURBES THETSM=F(STR)
C***** AUX ECHELLES NASA
REAL*8 THETSM(1),STR(1)
REAL*8 RNZDIA,VJET
DIMENSION XMOD(1),LIB1(1),LIB2(1)
C*****
COMMON /IDENT/NP01
COMMON /TITRE/LIB(10),IDATE(3)
C*****
DIMENSION NAM(7),IP0INT(2),IDIA(3),IV(4),IFIG(1)
DIMENSION NASA(5)
DIMENSION IVALY(6)
DIMENSION ITHE(4),IST(2)
DIMENSION X(2),Y(2)
DIMENSION VALX(5),XX(5)
DIMENSION STHETA(33),SSTR(33)
C*****
DATA NAM/'PA-J.BRASSEUR ECH.N.A.S.A.'/
DATA CX/300./,CY/210./
DATA IP0INT/'P0INT '/,IV/'VJET(FT/SEC)= '/
DATA IDIA/'DIA.(FT)= '/,IFIG/'FIG.'/
DATA NASA/'N.A.S.A./O.N.E.R.A.'/
DATA IVALY/60,80,100,120,140,160/
DATA ITHE/'THETA-S DU PIC '/
DATA IST/'STROUHAL'/
DATA VALX/+0.01,+0.10,+1.00,+10.0,+20.0/
DATA XX/16.,81.5,147.,212.5,232.5/
C***** TRACE DU CADRE ET DU CARTOUCHE
XT=10.
YT=200.
ANG=0.
NC=28
CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
KODE=1
NCL=40
NC=1
CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
KPLANC=KPLANC+1
C***** ECRITURE NASA
XT=CX-10.
YT=CY-55.
HL=3.
HH=5.
ANG=270.
NC=19
CALL KSTEP(1,1,4,1,XMOD)
CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
CALL KSTEP(1,1,1,1,XMOD)
C***** ECRITURE DANS LA MARGE DE LIB1 ET LIB2
HL=2.5
HH=2.5
NC=80
ANG=0.
XT=100.
YT=CY-6.5
CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
YT=CY-16.5

```



```

      CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
C***** ECRITURE DE POINT
      XT=10.
      YT=CY-30.
      NC=5
      HL=3.
      HH=3.
      ANG=0.
      CALL KTEXTE(XT,YT,HL,HH,IP0INT,ANG,NC,XMOD)
C*****
      XT=XT+18.
      NC=4
      CALL KFIXE(XT,YT,HL,HH,NP0I,ANG,NC,XMOD)
C***** ECRITURE DE VJET
      XT=50.
      NC=13
      CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
C*****
      XT=XT+42.
      NC=6
      S=SNGL(VJET)
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE DIA.(FT)=
      XT=120.
      NC=9
      CALL KTEXTE(XT,YT,HL,HH,IDIA,ANG,NC,XMOD)
C*****
      S=SNGL(RNZDIA)
      XT=XT+30.
      NC=5
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C***** ECRITURE DE FIG.
      XT=200.
      NC=4
      CALL KTEXTE(XT,YT,HL,HH,IFIG,ANG,NC,XMOD)
C***** TRACE DE L AXE VERTICAL
      XT=20.
      YT=20.
      ANG=90.
      XMIN=1.
      NC=0
      PAS=1.
      NGRAD=12
      NF=0
      PASMM=13.
      CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR L AXE VERTICAL
      XT=10.
      HL=2.
      HH=2.
      YT=-6.
      ANG=0.
      NC=3
      DO 100 I=1,6
      YT=YT+26.
      CALL KFIXE(XT,YT,HL,HH,IVALY(I),ANG,NC,XMOD)
100 CONTINUE
C***** ECRITURE DE THETA-S DU PIC
      YT=YT+16.
      XT=10.

```

```

        NC=14
        CALL KTEXTE(XT,YT,HL,HH,ISTHE,ANG,NC,XMOD)
C***** TRACE DE L AXE HORIZONTAL
        X(1)=20.
        X(2)=236.5
        Y(1)=20.
        Y(2)=Y(1)
        CALL KTRACE(X,Y,2,XMOD)
C***** ECRITURE DE STROUHAL
        XT=240.5
        YT=20.
        HL=2.
        HH=2.
        ANG=0.
        NC=8
        CALL KTEXTE(XT,YT,HL,HH,IST,ANG,NC,XMOD)
C***** ECRITURE DES COTATIONS SUR L AXE HORIZONTAL
        YT=14.
        NC=5
        DO 110 I=1,5
        CALL KDECIM(XX(I),YT,HL,HH,VALX(I),ANG,NC,XMOD)
110 CONTINUE
C***** GRADUATIONS DE L AXE HORIZONTAL
        CXOBJ1=20.
        CXOBJ2=236.5
        CYOBJ1=17.5
        CYOBJ2=22.5
        CXSUJ1=ALOG10(0.01)
        CXSUJ2=ALOG10(20.)
        CYSUJ1=-2.5
        CYSUJ2=+2.5
        CALL DEPLAC(0.,XMOD)
        CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
        CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
C*****
        Y(1)=-2.5
        Y(2)=+2.5
        PASX=0.001
        DO 120 J=1,3
        PASX=PASX*10.
        DO 120 I=1,9
        X(1)=ALOG10(I*PASX)
        X(2)=X(1)
        CALL KTRACE(X,Y,2,XMOD)
120 CONTINUE
        X(1)=ALOG10(10.)
        X(2)=X(1)
        CALL KTRACE(X,Y,2,XMOD)
        X(1)=ALOG10(20.)
        X(2)=X(1)
        CALL KTRACE(X,Y,2,XMOD)
C***** TRACE DES COURBES EN POINTE
        CXOBJ1=20.
        CXOBJ2=236.5
        CYOBJ1=20.
        CYOBJ2=163.
        CXSUJ1=ALOG10(+0.01)
        CXSUJ2=ALOG10(+20.)
        CYSUJ1=60.
        CYSUJ2=170.

```

```

      CALL DEPLAC(0.,XM0D)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XM0D)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XM0D)
C***** CONVERSION EN SIMPLE PRECISION DE THETSM ET STR
      NBPT=0
      DO 130 I=1,NFREQ
      IF(STR(I).LT.(0.01).OR.STR(I).GT.(+20.)) GO TO 130
      NBPT=NBPT+1
      STHETA(NBPT)=SNGL(THETSM(I))
      SSTR(NBPT)=SNGL(STR(I))
      SSTR(NBPT)=ALOG10(SSTR(NBPT))
130 CONTINUE
C*****
      CALL KSTEP(-16,1,1,1,XM0D)
      CALL KPOINT(SSTR,STHETA,NBPT,XM0D)
      CALL KSTEP(1,1,1,1,XM0D)
C*****
      CCX=CX+20.
      CALL DEPLAC(CCX,XM0D)
      RETURN
      END
IEXEC INOMB,%KEY=TPEAKN

```

```

*PROGRAMME SOURCE P242
C   TRANSFERT DE LA BANDE NUMERIQUE SUR LE DISQUE
    DIMENSION MAT(50)
    NR=1
    ND=2
    10 CALL BUFFER IN (NR,1,MAT,50,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)
    IF (IBU.EQ.3) GO TO 500
C   ECRITURE SUR DISQUE
    WRITE(ND,101)(MAT(I),I=1,50)
    101 FORMAT(50A4)
    GO TO 10
    500 STOP
    END
*PROGRAMME SOURCE P244
C   LECTURE DE BANDE BLOCS EN BCD FORMATTES FORMAT VARIABLE
C   TRAITEMENT DE PLUSIEURS FICHIERS
C   ECRITURE SUR DISQUE BLOCS EN EBCDIC FORMAT FIXE
    DIMENSION NZTYPE(3),FREQ(30),ANGLES(50),RDIST(50),DECB(50,31)
    DIMENSION MAT(100),NOMFICH(5)
    NR=1
    ND=2
    NFICH=0
C   SAUT DE FICHIERS
    READ(105,102)NFS
    102 FORMAT(15)
    IF(NFS.EQ.0)GO TO 20
C
    10 CALL BUFFER IN (NR,1,MAT,01,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)
    IF (IBU.EQ.3) NFICH=NFICH+1
    IF(NFICH.EQ.NFS)GO TO 20
    GO TO 10
    20 NFICH=0
    300 READ(105,103,END=600)(NOMFICH(I),I=1,5)
    NFICH=NFICH+1
    103 FORMAT(5A4)
C   ASSIGNATION DYNAMIQUE DU FICHIER DISQUE
    CALL AWSM('DISK',ND,NOMFICH,'F',1024,124)
C   LECTURE D UN FICHIER
C
    30 CALL BUFFER IN (NR,1,MAT,34,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)
    IF (IBU.EQ.3)GO TO 30
    DECODE(128,250,MAT(3))(IPT,N,VTUN,VJET,TEMPR,RHUM,SIDE,PRESS,XMACH,
    *DIANZL,NZTYPE
    WRITE(ND,150)IPT,N,VTUN,VJET,TEMPR,RHUM,SIDE,PRESS,XMACH,DIANZL,NZ
    *TYPE
C
    CALL BUFFER IN (NR,1,MAT,92,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)
    DECODE(360,252,MAT(3))(FREQ(I),I=1,30)
    WRITE(ND,152)(FREQ(I),I=1,30)
C
    CALL BUFFER IN (NR,1,MAT,68,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)
    DECODE(264,254,MAT(3))(ANGLES(I),I=1,N)
    WRITE(ND,154)(ANGLES(I),I=1,N)
C
    CALL BUFFER IN (NR,1,MAT,68,IBU)
    CALL BUFFER CHECK (NR,1,IBU,MOT)

```

```

        DECODE(264,254,MAT(3))(RDIST(I),I=1,N)
        WRITE(ND,154)(RDIST(I),I=1,N)
        DO 180 J=1,N
C
        CALL BUFFER IN (NR,1,MAT,95,IBU)
        CALL BUFFER CHECK (NR,1,IBU,MOT)
        DECODE(372,254,MAT(3))(DECB(J,I),I=1,31)
180    WRITE(ND,154)(DECB(J,I),I=1,31)
        REWIND ND
        WRITE(108,104)NFICH
104    FORMAT(' FICHIER ',I2)
        GO TO 300
600    WRITE(108,101)
101    FORMAT(' TRAVAIL TERMINE')
150    FORMAT(13A4)
250    FORMAT(2I10,8F12.2,3A4)
152    FORMAT(30A4)
252    FORMAT(30F12.0)
154    FORMAT(31A4)
254    FORMAT(31F12.2)
        STOP
        END

```

```

*
C PROGRAMME SOURCE RNOISE1
C   ELIMINATION D UNE LIGNE DE MESURE MODANE
C   A PARTIR DE FICHIERS 2 LIGNES
C   FICHER UTILISE DANS LE CALCUL DE CHAMP LOINTAIN
    DIMENSION TAB1(50),TAB2(50),ITAB(6),NP01(15)
    EQUIVALENCE (TAB1(2),ITAB(1))
    DATA NBP1ST/8/
    KF=0
    ND1=1
    ND3=3
    10 READ(105,101,END=900)NBPT,(NP01(I),I=1,NBPT)
    101 FORMAT(16I5)
C   BOUCLE SUR LES POINTS
    DO 5 L=1,NBPT
    NUMER=1
    ND=ND1+KF*3
    REWIND ND
    K0DR=0
C   LECTURE BLOC DE TETE
    600 READ(ND,105,END=610)(TAB1(I),I=1,50)
    105 FORMAT(50A4)
    IF(ITAB(1).NE.NP01(L))GO TO 500
    INDEX=ITAB(4)
    IF(INDEX.EQ.3.OR.INDEX.EQ.4.OR.INDEX.EQ.7.OR.INDEX.EQ.8)NUMER=NUME
    *R+1;GO TO 500
    WRITE(ND3,105)(TAB1(I),I=1,50)
C   LECTURE BLOCS SPECTRE
    DO 20 IA=1,ITAB(5)
    READ(ND,105,END=630)(TAB2(I),I=1,50)
    20 WRITE(ND3,105)(TAB2(I),I=1,50)
    NUMER=NUMER+1
    GO TO 600
    500 DO 30 J=1,ITAB(5)
    30 READ(ND,105,END=620)(TAB2(I),I=1,50)
    IF(NUMER.LE.NBP1ST)GO TO 600
    GO TO 5
    610 K0DR=1
    WRITE(108,102)K0DR
    STOP
    620 K0DR=2
    WRITE(108,102)K0DR
    STOP
    630 K0DR=3
    WRITE(108,102)K0DR
    102 FORMAT(' K0DR',I5)
    STOP
    5 CONTINUE
    KF=KF+1
    GO TO 10
    900 REWIND ND3
    WRITE(108,103)
    103 FORMAT(' TRAVAIL TERMINE')
    STOP
    END

```

```

*
CPRPROGRAMME SOURCE RNOISE12
C   REGROUPEMENT DE 2 LIGNES DE MESURE MODANE
C   A PARTIR DE FICHIERS 1 LIGNE
C   FICHIER UTILISE DANS LE CALCUL DE LOCALISATION DE SOURCE
C   DIMENSION TAB1(50),TAB2(50),ITAB(6),NP01(15)
C   EQUIVALENCE (TAB1(2),ITAB(1))
C   DATA NBP1ST/4/
C   KF=0
C   ND1=1
C   ND2=2
C   ND3=3
C   10 READ(105,101,END=900)NBPT,(NP01(I),I=1,NBPT)
C   101 FORMAT(16I5)
C   BOUCLE SUR LES POINTS
C   DO 5 L=1,NBPT
C   BOUCLE SUR LES LIGNES
C   DO 5 IL=1,2
C   NUMER=1
C   IF(IL.EQ.1)ND=ND1+KF*3
C   IF(IL.EQ.2)ND=ND2+KF*3
C   REWIND ND
C   50 K0DR=0
C   LECTURE BLOC DE TETE
C   600 READ(ND,105,END=610)(TAB1(I),I=1,50)
C   105 FORMAT(50A4)
C   IF(ITAB(1).NE.NP01(L))GO TO 500
C   WRITE(ND3,105)(TAB1(I),I=1,50)
C   LECTURE BLOCS SPECTRE
C   DO 20 IA=1,ITAB(5)
C   READ(ND,105,END=630)(TAB2(I),I=1,50)
C   20 WRITE(ND3,105)(TAB2(I),I=1,50)
C   NUMER=NUMER+1
C   IF(NUMER.LE.NBP1ST)GO TO 50
C   GO TO 5
C   500 DO 30 J=1,ITAB(5)
C   30 READ(ND,105,END=620)(TAB2(I),I=1,50)
C   GO TO 600
C   610 K0DR=1
C   WRITE(108,102)K0DR
C   STOP
C   620 K0DR=2
C   WRITE(108,102)K0DR
C   STOP
C   630 K0DR=3
C   WRITE(108,102)K0DR
C   102 FORMAT(' K0DR',15)
C   STOP
C   5 CONTINUE
C   KF=KF+1
C   GO TO 10
C   900 REWIND ND3
C   WRITE(108,103)
C   103 FORMAT(' TRAVAIL TERMINE')
C   STOP
C   END

```

```

*
CPROGRAMME SOURCE ECHN4LM
  REAL*8 DUMP(3446), DUMP1(132), DUMP2(74), DUMP3(1800), DUMP4(3500)
  REAL*8 XOVERD, THETAS, XDMAX, THETSM
  REAL*8 RNZDIA, VJET, FREQ
  REAL*8 STR, DFMAX, XPEAKN, XPEAKF
  REAL*8 PSIVAL, XDVAL, ST, STP, VJ3, VJ4, VAMB
  REAL*8 PSIS, XDN
  COMMON /IDENT/NP01
  COMMON /TITRE/LIB(10), IDATE(3)
  COMMON /TABD/STR(33), DFMAX(33), XPEAKN(33), XPEAKF(33)
  COMMON /SUB/VJ3, FREQ(33), THETAS(50, 33), XOVERD(50, 33), NTAB(33),
1    NTEST, XDMAX(33), THETSM(33), RNZDIA, NZTYPE(10), NFREQ
  COMMON/SUB2/ST(35), STP(35), IWT1, IWT2, VAMB, VJ4
  COMMON/SUB3/PSIVAL(50), XDVAL(50, 35)
  COMMON/SUB4/PSIS(50, 35), XDN(50, 35)
  DIMENSION LIB1(20), LIB2(20), LIBE(10), IDAT(3)
  DIMENSION XM0D(44)
C*****
  EQUIVALENCE (DUMP(1), VJ3)
  EQUIVALENCE (DUMP1(1), STR(1))
  EQUIVALENCE (DUMP2(1), ST(1))
  EQUIVALENCE (DUMP3(1), PSIVAL(1))
  EQUIVALENCE (DUMP4(1), PSIS(1, 1))
C***** PREPARATION DU TRACE
  DO 100 I=1, 44
    XM0D(I)=0.
  100 CONTINUE
  NDT=69
  CALL OPENTR(NDT, XM0D)
  KPLANC=0
C***** LECTURE DES DONNEES NOISE4
  READ(7) DUMP2
  READ(7) DUMP3
  READ(7) NP01
  READ(7) LIB
  READ(7) IDATE
  REWIND 7
C***** LECTURE DES DONNEES NOISE3
  READ(8) DUMP
  READ(8) DUMP1
  READ(8) NP0
  READ(8) LIBE
  READ(8) IDAT
  READ(8) LIB1
  READ(8) LIB2
  REWIND 8
C***** TRACE
  DO 10 J=1, 33
    DO 10 I=1, 50
      THETAS(I, J)=PSIVAL(I)
10  XOVERD(I, J)=XDVAL(I, J)
  DO 20 J=1, 33
    20  NTAB(J)=50
  VJET=VJ3
  CALL TXSDN4(KPLANC, XOVERD, THETAS, NTAB, NFREQ, XDMAX, THETSM, RNZDIA,
1  VJET, FREQ, LIB1, LIB2, XM0D)
C***** FIN DE TRACE
  CALL CLOSTR(XM0D)
  WRITE(108, 2010) KPLANC

```



```

2010 FORMAT(///3X,'NB. PLANCHES=' ,I3//)
      STOP
      END
      SUBROUTINE TXSDN4(KPLANC,XOVERD,THETAS,NTAB,NFREQ,XDMAX,
1THETSM,RNZDIA,VJET,FREQ,LIB1,LIB2,XMOD)
C***** SOUS-PROGRAMME DE TRACE DES COURBES X/D=F(P.S.I.S)
      REAL*8 XOVERD(50,33),THETAS(50,33),XDMAX(33),THETSM(33)
      REAL*8 RNZDIA,VJET,FREQ(33),ST,STP
      COMMON /IDENT/NP01
      COMMON /TITRE/LIB(10),IDATE(3)
      COMMON /TABD/STR(33),DFMAX(33),XPEAKN(33),XPEAKF(33)
      COMMON/SUB2/ST(35),STP(35)
      REAL*8 STR,DFMAX,XPEAKN,XPEAKF
      DIMENSION XMOD(1),NTAB(1),LIB1(1),LIB2(1)
C*****
      DIMENSION NAM(7),NASA(5),IPOINT(2),IDEG(1)
      DIMENSION IV(4)
      DIMENSION ITHE(2),IXSD(1),IFREQ(1)
      DIMENSION IANG(7),XIANG(7),IVALX(6)
      DIMENSION SXSD(50),STHETA(50)
      DIMENSION IFIG(1)
      DIMENSION X12(2),Y12(2)
C*****
      DATA NAM/'PA-J.BRASSEUR   ECH.N.A.S.A.'/
      DATA NASA/'N.A.S.A./O.N.E.R.A. '/
      DATA IPOINT/'POINT  '/
      DATA IDEG/'DEG  '/
      DATA IV/' V0 (FT/SEC)=  '/
      DATA IST1/'ST0='/',IST2/'ST ='/',IST3/'STP='/'
      DATA ITHE/' PSI-S  '/,IXSD/'X/D  '/,IFREQ/'FREQ'/
C*****
      DATA CX/210./,CY/300./
      DATA IANG/40,60,80,100,120,140,160/
      DATA XIANG/47.25,67.75,88.25,108.75,129.25,149.75,170.25/
      DATA IVALX/0,2,4,6,8,10/
      DATA IFIG/'FIG. '/
C*****
C***** TRACE DE 2 COURBES PAR PLANCHE
      NBCT=0
      DO 210 IF=1,NFREQ
      IF(NTAB(IF).EQ.0) GO TO 210
      IF(NBCT.EQ.1) GO TO 160
C***** TRACE DU CADRE ET DU CARTOUCHE
      IF(NBCT.EQ.2) NBCT=0
      XT=10.
      YT=10.
      ANG=90.
      NC=28
      CALL TPL(CX,CY,XT,YT,ANG,NAM,NC,XMOD)
      KODE=0
      NCL=40
      NC=1
      CALL TCART(KODE,CX,CY,LIB,NCL,IDATE,NC,XMOD)
      KPLANC=KPLANC+1
C***** ECRITURE NASA
      XT=55.
      YT=CY-10.
      HL=3.
      HH=5.
      ANG=0.

```

```

NC=19
CALL KSTEP(1,1,4,1,XMOD)
CALL KTEXTE(XT,YT,HL,HH,NASA,ANG,NC,XMOD)
CALL KSTEP(1,1,1,1,XMOD)
C***** ECRITURE DANS LA MARGE DE LIB1,LIB2
HL=2.5
HH=2.5
NC=80
ANG=90.
XT=6.5
YT=100.
CALL KTEXTE(XT,YT,HL,HH,LIB1,ANG,NC,XMOD)
XT=16.5
CALL KTEXTE(XT,YT,HL,HH,LIB2,ANG,NC,XMOD)
C***** ECRITURE NPOI
XT=25.
YT=CY-28.
NC=5
HL=3.
ANG=0.
HH=3.
CALL KTEXTE(XT,YT,HL,HH,IPOINT,ANG,NC,XMOD)
C*****
XT=XT+18.
NC=4
CALL KFIXE(XT,YT,HL,HH,NPOI,ANG,NC,XMOD)
C***** ECRITURE DE FIG.
XT=180.
NC=4
CALL KTEXTE(XT,YT,HL,HH,IFIG,ANG,NC,XMOD)
C***** TRACE DU PREMIER AXE VERTICAL(PARTIE HAUTE DE LA PLANCHE)
XT=40.
YT=130.
NC=0
PASMM=20.5
ANG=90.
XMIN=1.
PAS=1.
NGRAD=6
NF=0
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE VERTICAL(PARTIE BASSE DE LA PLANCHE)
YT=10.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU PREMIER AXE HORIZONTAL
XT=40.
YT=130.
ANG=0.
PASMM=10.25
NGRAD=15
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** TRACE DU SECOND AXE HORIZONTAL
YT=10.
CALL KAXE(XT,YT,NORD,NC,PASMM,ANG,XMIN,PAS,NGRAD,NF,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES HORIZONTAUX
YT=5.
ANG=0.
HL=2.
HH=2.
NC=3

```

```

      DO 120 I=1,7
      XT=XIANG(I)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
120  CONTINUE
      XT=XT+10.25
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
      YT=10.
      XT=186.5
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
      YT=112.
      XT=100.
      NC=13
      CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
      XT=60.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IST1,ANG,NC,XMOD)
      YT=YT-7.
      CALL KTEXTE(XT,YT,HL,HH,IST2,ANG,NC,XMOD)
      YT=YT-7.
      CALL KTEXTE(XT,YT,HL,HH,IST3,ANG,NC,XMOD)
C*****
      YT=125.
      NC=3
      DO 130 I=1,7
      XT=XIANG(I)
      CALL KFIXE(XT,YT,HL,HH,IANG(I),ANG,NC,XMOD)
130  CONTINUE
      XT=XT+10.25
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IDEG,ANG,NC,XMOD)
      YT=YT+5.
      XT=186.5
      NC=7
      CALL KTEXTE(XT,YT,HL,HH,ITHE,ANG,NC,XMOD)
      YT=240.
      XT=100
      NC=13
      CALL KTEXTE(XT,YT,HL,HH,IV,ANG,NC,XMOD)
      XT=60.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IST1,ANG,NC,XMOD)
      YT=YT-7.
      CALL KTEXTE(XT,YT,HL,HH,IST2,ANG,NC,XMOD)
      YT=YT-7.
      CALL KTEXTE(XT,YT,HL,HH,IST3,ANG,NC,XMOD)
C***** ECRITURE DES COTATIONS SUR LES AXES VERTICAUX
      XT=32.
      NC=2
      YT=-10.5
      DO 140 I=1,6
      YT=YT+20.5
      CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
140  CONTINUE
      XT=21.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C*****
      XT=32.

```

```

      YT=109.5
      NC=2
      DO 150 I=1,6
      YT=YT+20.5
      CALL KFIXE(XT,YT,HL,HH,IVALX(I),ANG,NC,XMOD)
150  CONTINUE
      XT=21.
      NC=3
      CALL KTEXTE(XT,YT,HL,HH,IXSD,ANG,NC,XMOD)
C***** CONVERSION EN SIMPLE PRECISION DE X/D ET THETA
160  CONTINUE
      DO 170 I=1,NTAB(IF)
      SXSD(I)=SNGL(XOVERD(I,IF))
      STHETA(I)=SNGL(THETAS(I,IF))
170  CONTINUE
      NBCT=NBCT+1
C*****
      IF(NBCT.EQ.1) YT=240. ; GO TO 190
      YT=112.
190  CONTINUE
C***** ECRITURE VJET
      XT=140.
      NC=6
      HL=2.
      HH=2.
      S=SNGL(VJET)
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****STROUHAL STATIQ. ET DYNAMIQ.
      XT=70.
      NC=7
      S=SNGL(STR(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      YT=YT-7.
      S=SNGL(ST(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
      YT=YT-7.
      S=SNGL(STP(IF))
      CALL KDECIM(XT,YT,HL,HH,S,ANG,NC,XMOD)
C*****
C***** TRACE EN POINTE DES COURBES
      CXSUJ1=30.
      CXSUJ2=170.
      CYSUJ1=-0.5
      CYSUJ2=10.
      CXOBJ1=40.
      CXOBJ2=183.5
      IF(NBCT.EQ.1) CYOBJ1=124.875 ; CYOBJ2=232.5 ; GO TO 200
      CYOBJ1=4.875
      CYOBJ2=112.5
200  CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(CXSUJ1,CXSUJ2,CYSUJ1,CYSUJ2,XMOD)
      CALL DIMOBJ(CXOBJ1,CXOBJ2,CYOBJ1,CYOBJ2,XMOD)
      NBPT=NTAB(IF)
      CALL KSTEP(-16,1,1,1,XMOD)
      CALL KPPOINT(STHETA,SXSD,NBPT,XMOD)
      CALL KSTEP(1,1,1,1,XMOD)
      IF(NBCT.NE.2) GO TO 220
      CCX=CCX+20.
      CALL DEPLAC(CCX,XMOD)

```

```

      GO TO 210
220  CONTINUE
      CALL DEPLAC(0.,XMOD)
      CALL DIMSUJ(0.,CX,0.,CY,XMOD)
      CALL DIMOBJ(0.,CX,0.,CY,XMOD)
210  CONTINUE
      IF(NBCT.EQ.2) RETURN
      CCX=CX+20.
      CALL DEPLAC(CCX,XMOD)
      RETURN
      END

```




Report Documentation Page

1. Report No. NASA TP-3053		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle J-85 Jet Engine Noise Measured in the ONERA S1 Wind Tunnel and Extrapolated to Far Field				5. Report Date January 1991	
				6. Performing Organization Code	
7. Author(s) Paul T. Soderman, Alain Julienne,* and Adolph Atencio, Jr.				8. Performing Organization Report No. A-89265	
				10. Work Unit No. 307-50-81	
9. Performing Organization Name and Address Ames Research Center Moffett Field, CA 94035-1000				11. Contract or Grant No.	
				13. Type of Report and Period Covered Technical Publication	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001				14. Sponsoring Agency Code	
15. Supplementary Notes Point of Contact: Paul T. Soderman, Ames Research Center, MS 247-2 Moffett Field, CA 94035-1000, (415) 604-6675 or FTS 464-6675 *Office National d'Études et de Recherches Aérospatiales, Châtillon (Hauts-de-Seine), France					
16. Abstract <p>Noise from a J-85 turbojet with a conical, convergent nozzle was measured in simulated flight in the ONERA S1 Wind Tunnel. Data are presented for several flight speeds up to 130 m/sec and for radiation angles of 40° to 160° relative to the upstream direction. The jet was operated with subsonic and sonic exhaust speeds. A moving microphone on a 2-m sideline was used to survey the radiated sound field in the acoustically treated, closed test section. The data were extrapolated to a 122-m sideline by means of a multiple-sideline source-location method, which was used to identify the acoustic source regions, directivity patterns, and near field effects. The source-location method is described along with its advantages and disadvantages.</p> <p>Results indicate that the effects of simulated flight on J-85 noise are significant. At the maximum forward speed of 130 m/sec, the peak overall sound levels in the aft quadrant were attenuated approximately 10 dB relative to sound levels of the engine operated statically. As expected, the simulated flight and static data tended to merge in the forward quadrant as the radiation angle approached 40°. There is evidence that internal engine or shock noise was important in the forward quadrant. The data are compared with published predictions for flight effects on pure jet noise and internal engine noise. A new empirical prediction is presented that relates the variation of internally generated engine noise or broadband shock noise to forward speed. Measured near field noise extrapolated to far field agrees reasonably well with data from similar engines tested statically outdoors, in flyover, in a wind tunnel, and on the Bertin Aerotrain. Anomalies in the results for the forward quadrant and for angles above 140° are discussed.</p> <p>The multiple-sideline method proved to be cumbersome in this application, and it did not resolve all of the uncertainties associated with measurements of jet noise close to the jet. The simulation was complicated by wind-tunnel background noise and the propagation of low-frequency sound around the circuit.</p>					
17. Key Words (Suggested by Author(s)) Aeroacoustics Jet noise Wind tunnel noise Jet engine noise			18. Distribution Statement Unclassified-Unlimited Subject Category - 71		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 180	
				22. Price A04	

**National Aeronautics and
Space Administration
Code NTT-4**

**Washington, D.C.
20546-0001**

**Official Business
Penalty for Private Use, \$300**

**SPECIAL FOURTH-CLASS RATE
POSTAGE & FEES PAID
NASA
Permit No. G-27**

NASA

**POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return**
